

US009736054B2

(12) **United States Patent**
Bacchu et al.

(10) **Patent No.:** **US 9,736,054 B2**
(45) **Date of Patent:** **Aug. 15, 2017**

(54) **MULTICAST ACTIVE SOURCE DISCOVERY AND MANAGEMENT FOR LAYER-2 INTERCONNECT SOLUTIONS**

(75) Inventors: **Nataraj Bacchu**, Sunnyvale, CA (US); **Ashok Chippa**, Mountain View, CA (US); **Hasmit Grover**, Fremont, CA (US); **Raghava Sivaramu**, Sunnyvale, CA (US); **Dino Farinacci**, San Jose, CA (US)

(73) Assignee: **Cisco Technology, Inc.**, San Jose, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1156 days.

(21) Appl. No.: **13/253,505**

(22) Filed: **Oct. 5, 2011**

(65) **Prior Publication Data**

US 2013/0089093 A1 Apr. 11, 2013

(51) **Int. Cl.**

H04L 12/28 (2006.01)
H04L 12/715 (2013.01)
H04L 12/46 (2006.01)
H04L 12/761 (2013.01)

(52) **U.S. Cl.**

CPC **H04L 45/04** (2013.01); **H04L 12/4633** (2013.01); **H04L 12/4641** (2013.01); **H04L 45/16** (2013.01); **H04L 45/64** (2013.01); **H04L 2212/00** (2013.01)

(58) **Field of Classification Search**

USPC 370/324, 329, 390
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

7,830,787	B1	11/2010	Wijnands et al.
2002/0091846	A1	7/2002	Garcia-Luna-Aceves et al.
2007/0104194	A1	5/2007	Wijnands et al.
2007/0217428	A1	9/2007	Wijnands et al.
2008/0072035	A1*	3/2008	Johnson H04L 63/0428 713/153
2008/0298360	A1	12/2008	Wijnands et al.
2009/0037607	A1*	2/2009	Farinacci H04L 12/4641 709/249
2009/0175274	A1	7/2009	Aggarwal et al.
2010/0061269	A1*	3/2010	Banerjee H04L 12/4675 370/254
2010/0094862	A1*	4/2010	Bent G06F 17/30545 707/716
2011/0216685	A1	9/2011	Kish et al.
2011/0243131	A1	10/2011	Amir et al.

(Continued)

OTHER PUBLICATIONS

NPL document named "Overlay Transport Virtualization", draft-hasmit-otv-01, Published by: Network Working Group, on Oct. 24, 2010.*

(Continued)

Primary Examiner — Gregory Sefcheck

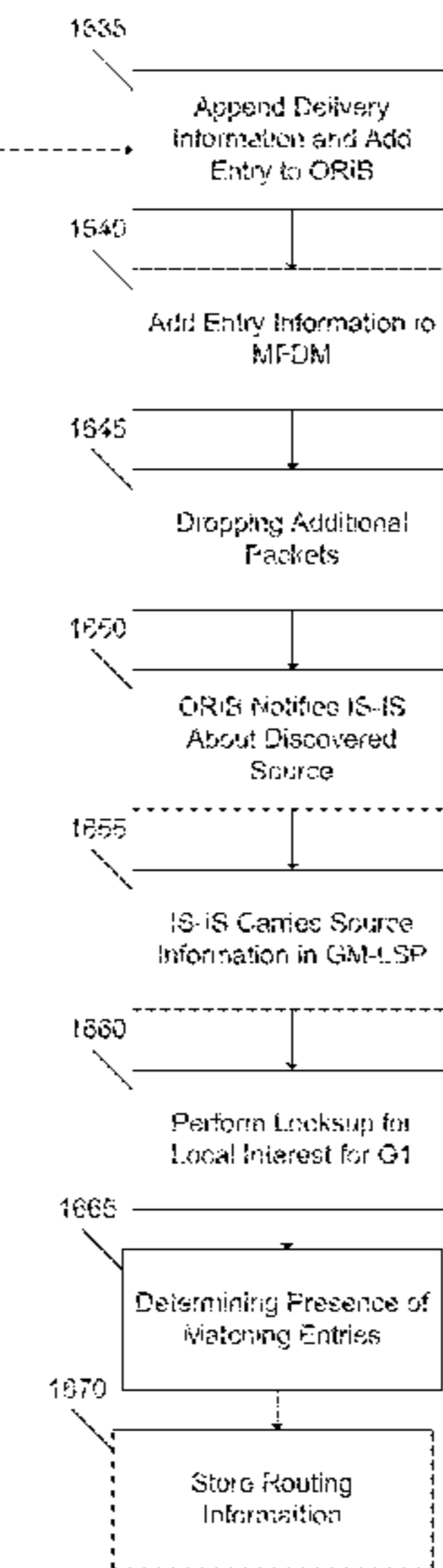
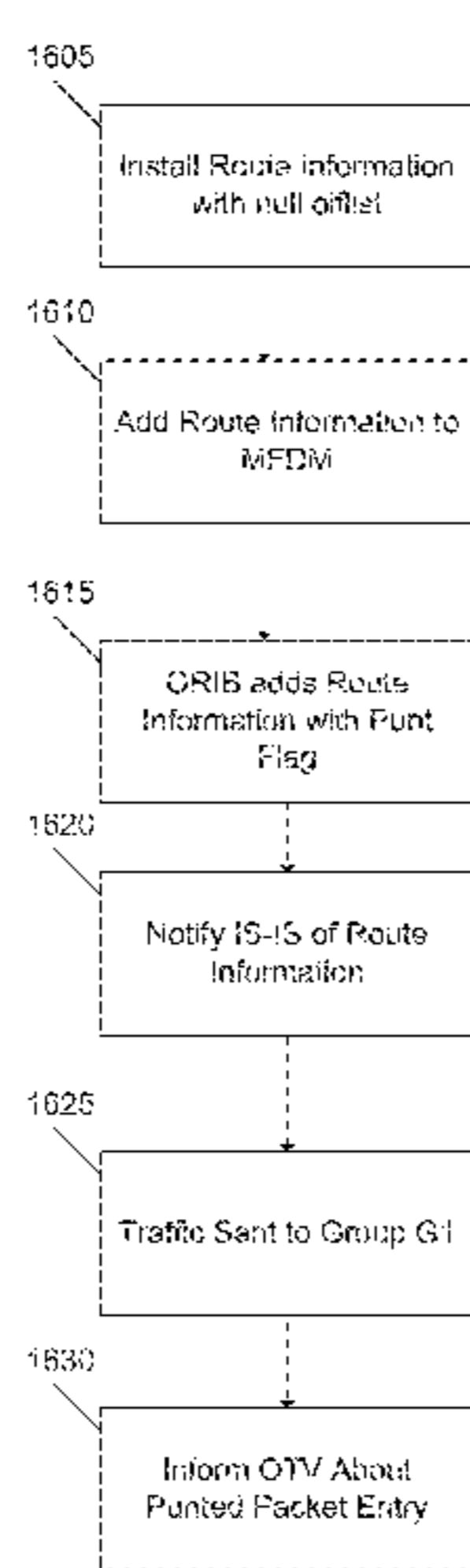
Assistant Examiner — Majid Esmailian

(74) *Attorney, Agent, or Firm* — Merchant & Gould P.C.

(57) **ABSTRACT**

Methods and systems may be provided for installing a route entry associated with multicast traffic to a memory. Client devices may be notified of the route entry for advertisement by an active source device. The delivery group and delivery source may be retrieved from the information for the route entry. Multicast data trees may maintain delivery group and delivery source information for access.

18 Claims, 22 Drawing Sheets



(56)

References Cited

U.S. PATENT DOCUMENTS

2013/0021942 A1 1/2013 Bacthu et al.

OTHER PUBLICATIONS

U.S. Office Action mailed Mar. 27, 2013 in U.S. Appl. No. 13/185,326, 14 pages.

“Central Processing Unit,” Pfaffenberger, Bryan. Webster’s New World Computer Dictionary, Ninth Edition. New York: Hungry Minds, Inc., 2001, Print. 2 pages.

U.S. Office Action mailed Sep. 13, 2013 in U.S. Appl. No. 13/185,236, 17 pages.

U.S. Office Action mailed Jul. 18, 2014 in U.S. Appl. No. 13/185,236, 12 pgs.

U.S. Appl. No. 13/185,326, filed Jul. 18, 2011 entitled Granular Control of Multicast Delivery Services for Layer-2 Interconnect Solutions, Inventors: Bacthu et al.

U.S. Office Action mailed Feb. 5, 2015 in U.S. Appl. No. 13/185,236, 15 pgs.

* cited by examiner

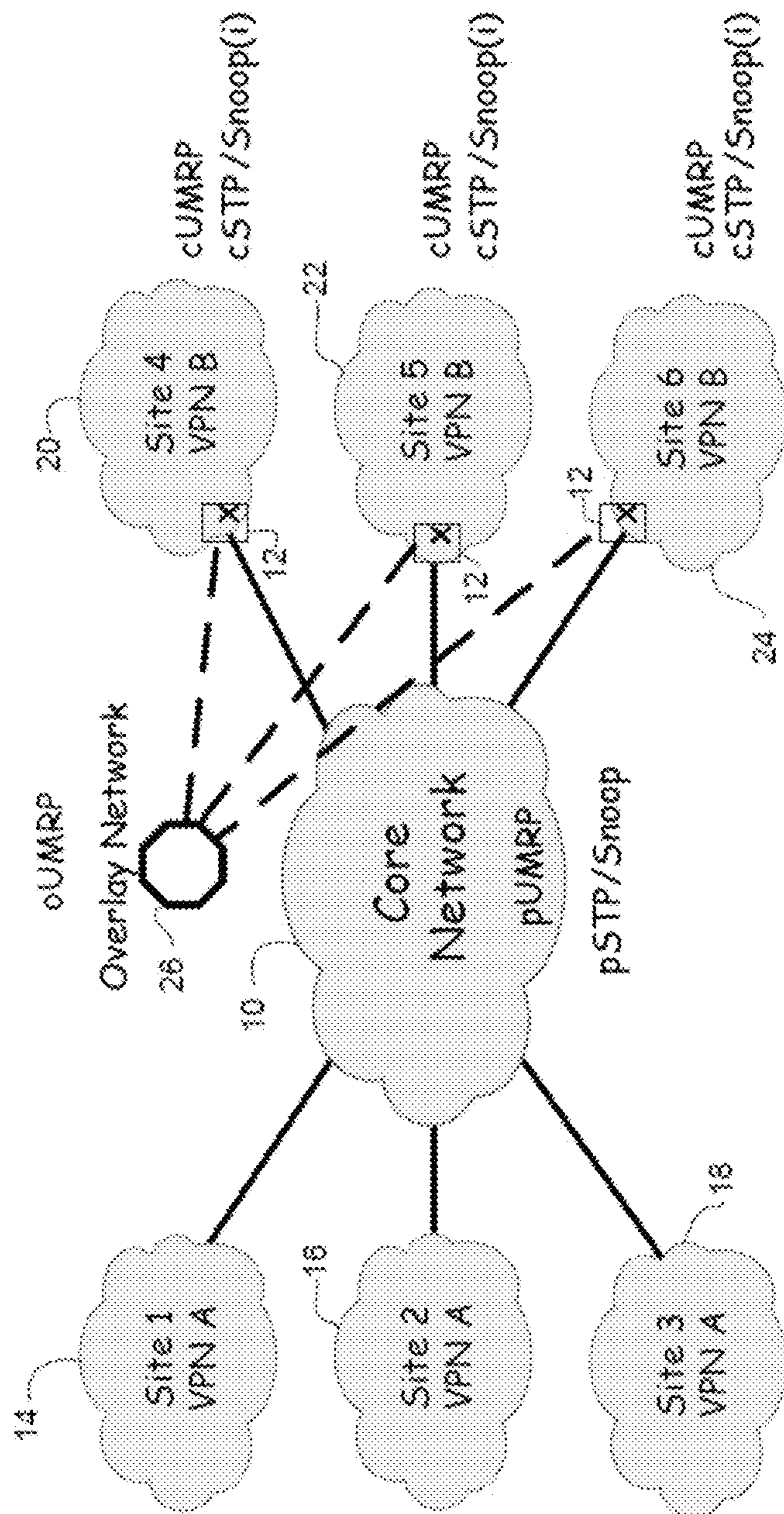


FIGURE 1

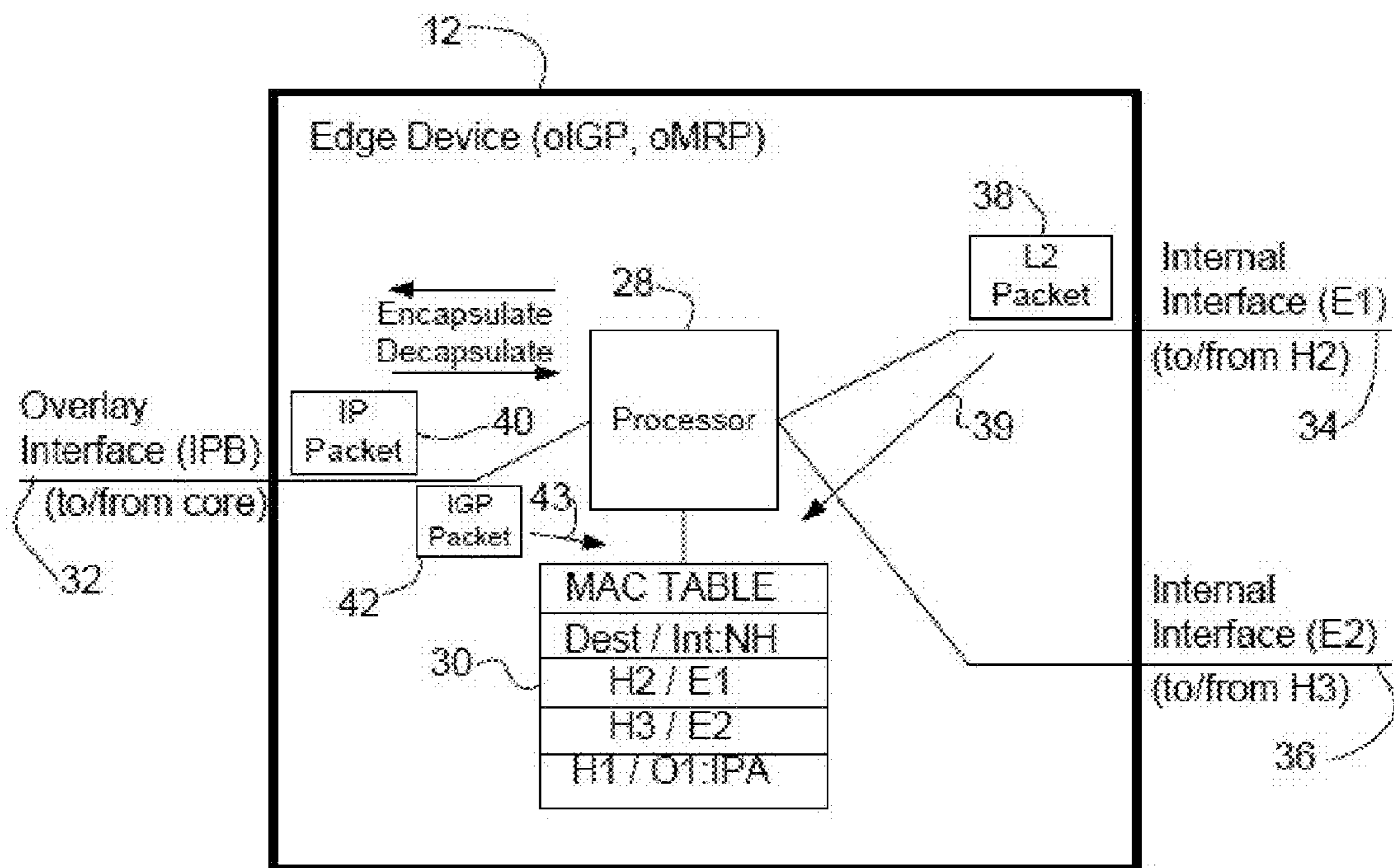


FIGURE 2

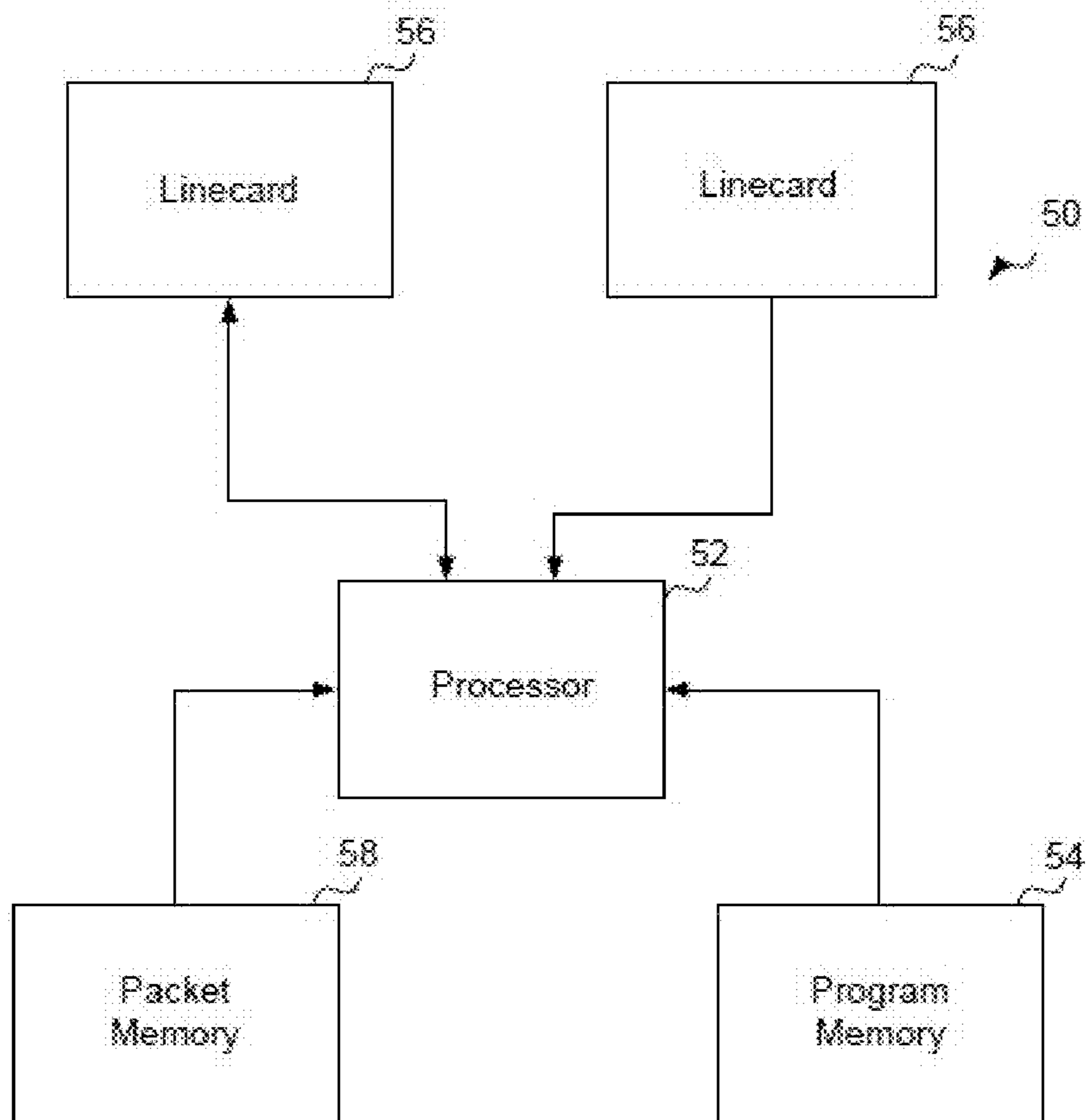


FIGURE 3

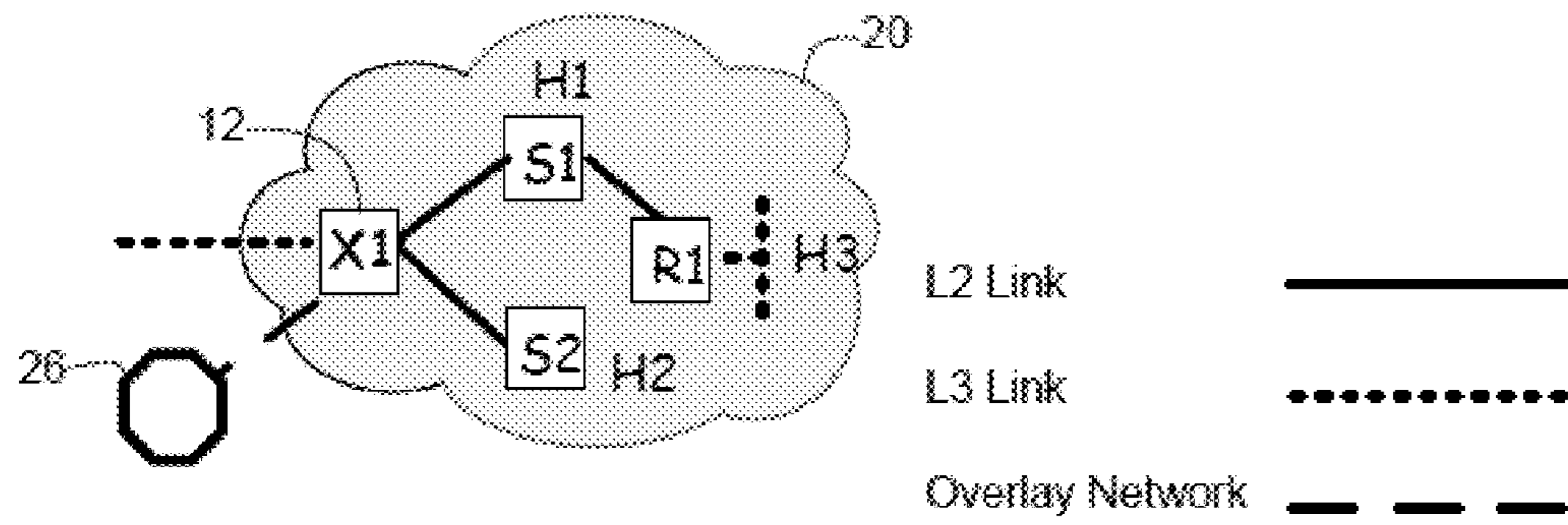


FIGURE 4A

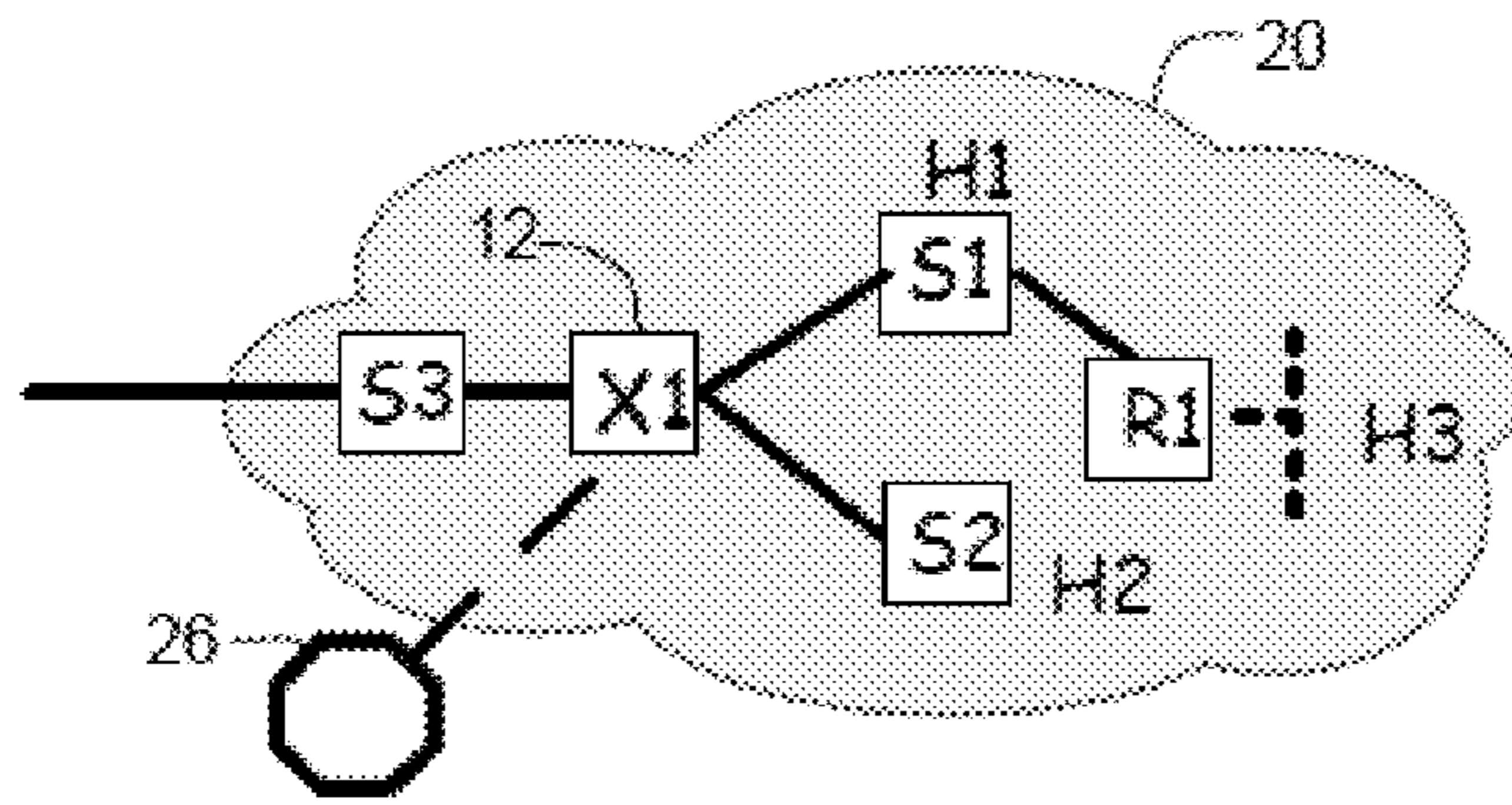


FIGURE 4B

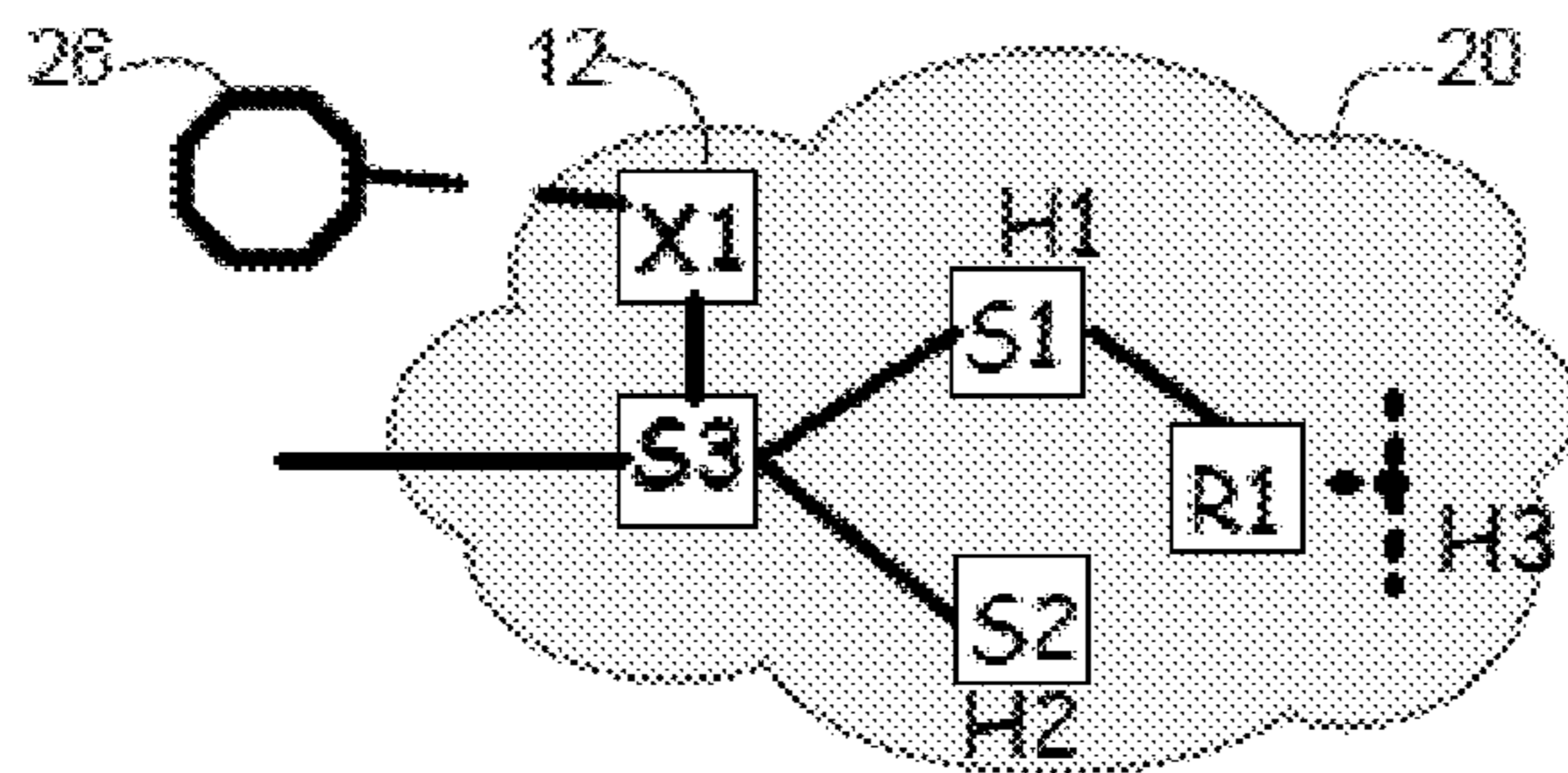


FIGURE 4C

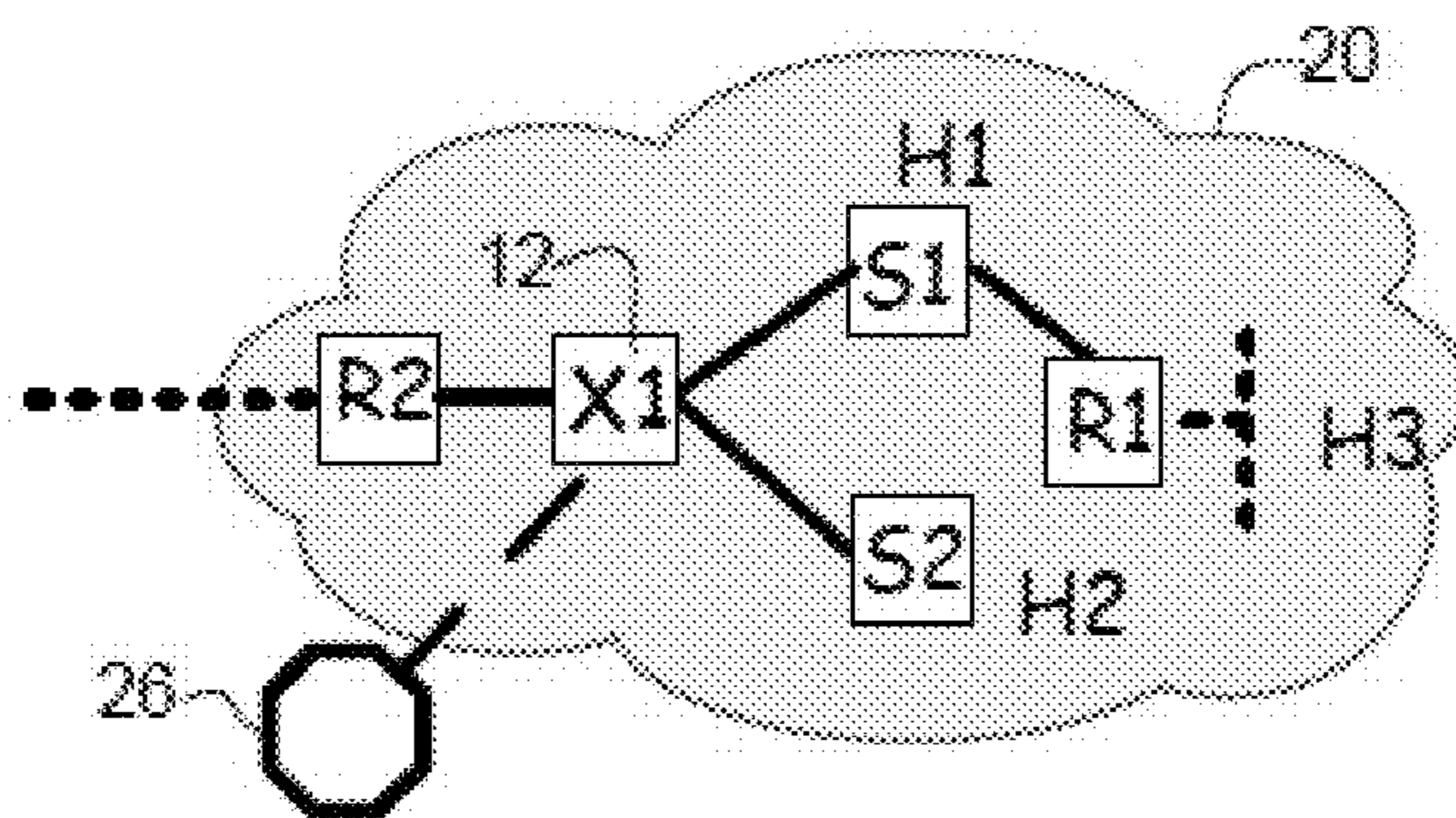


FIGURE 4D

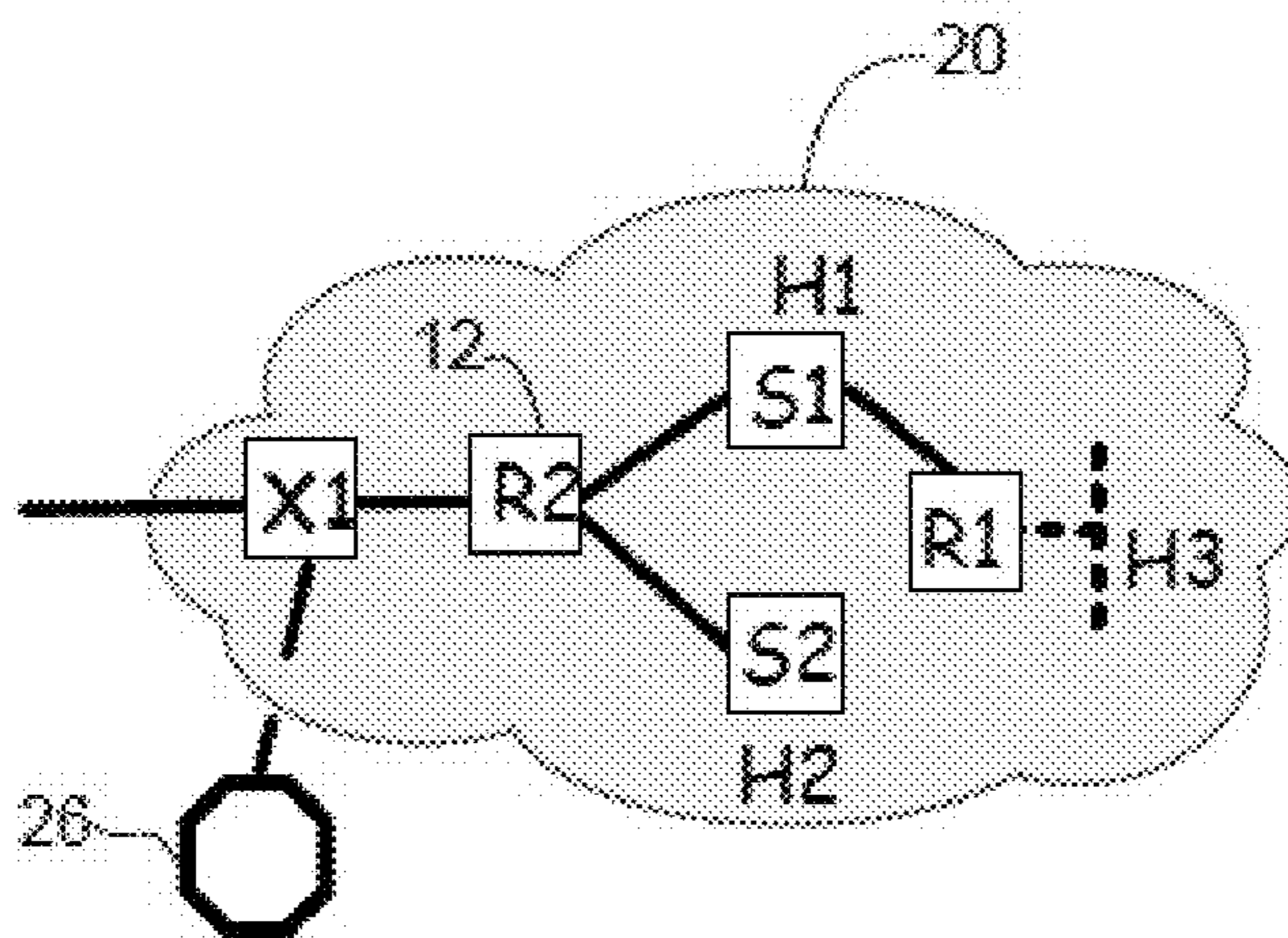


FIGURE 4E

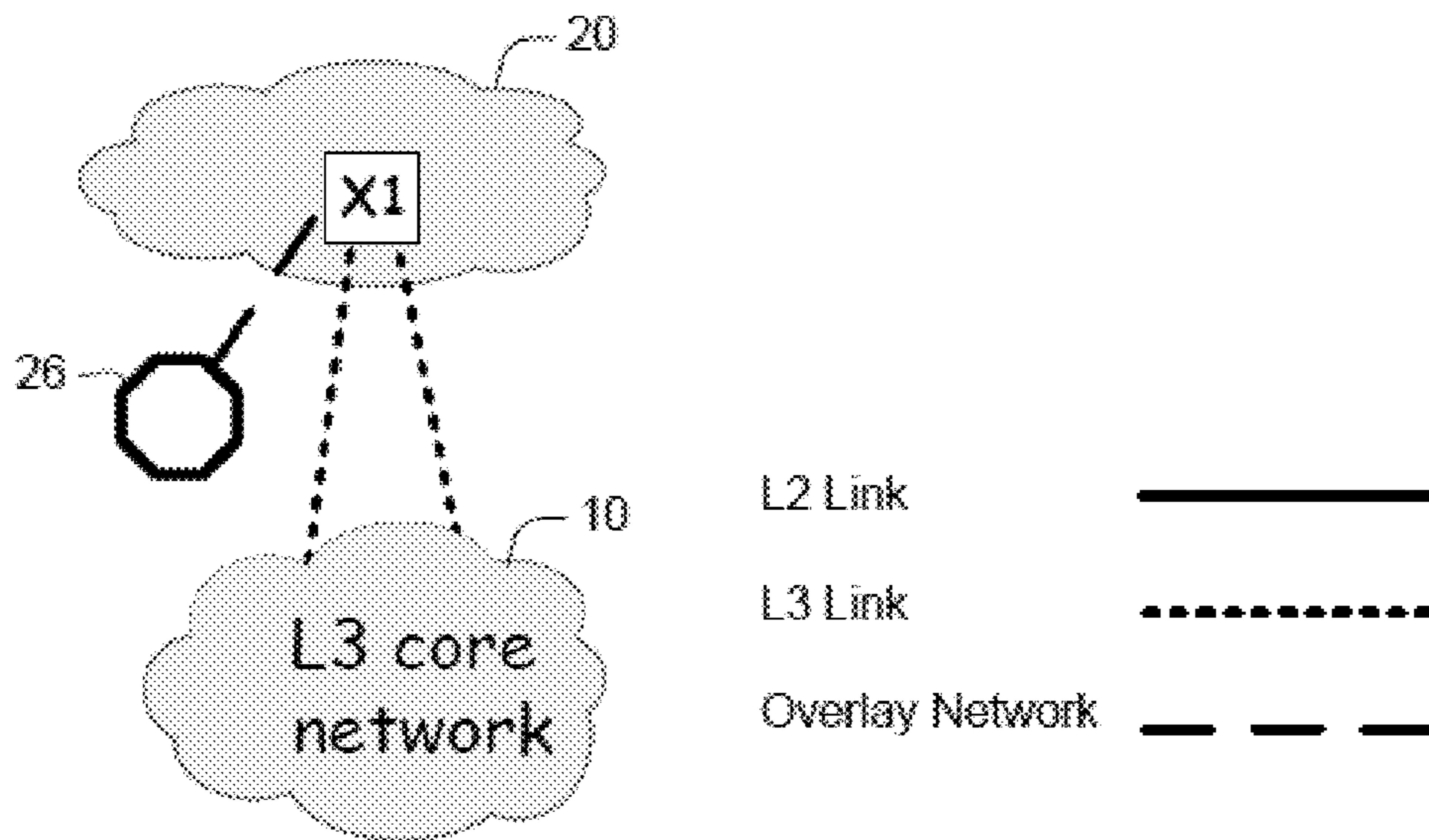


FIGURE 5A

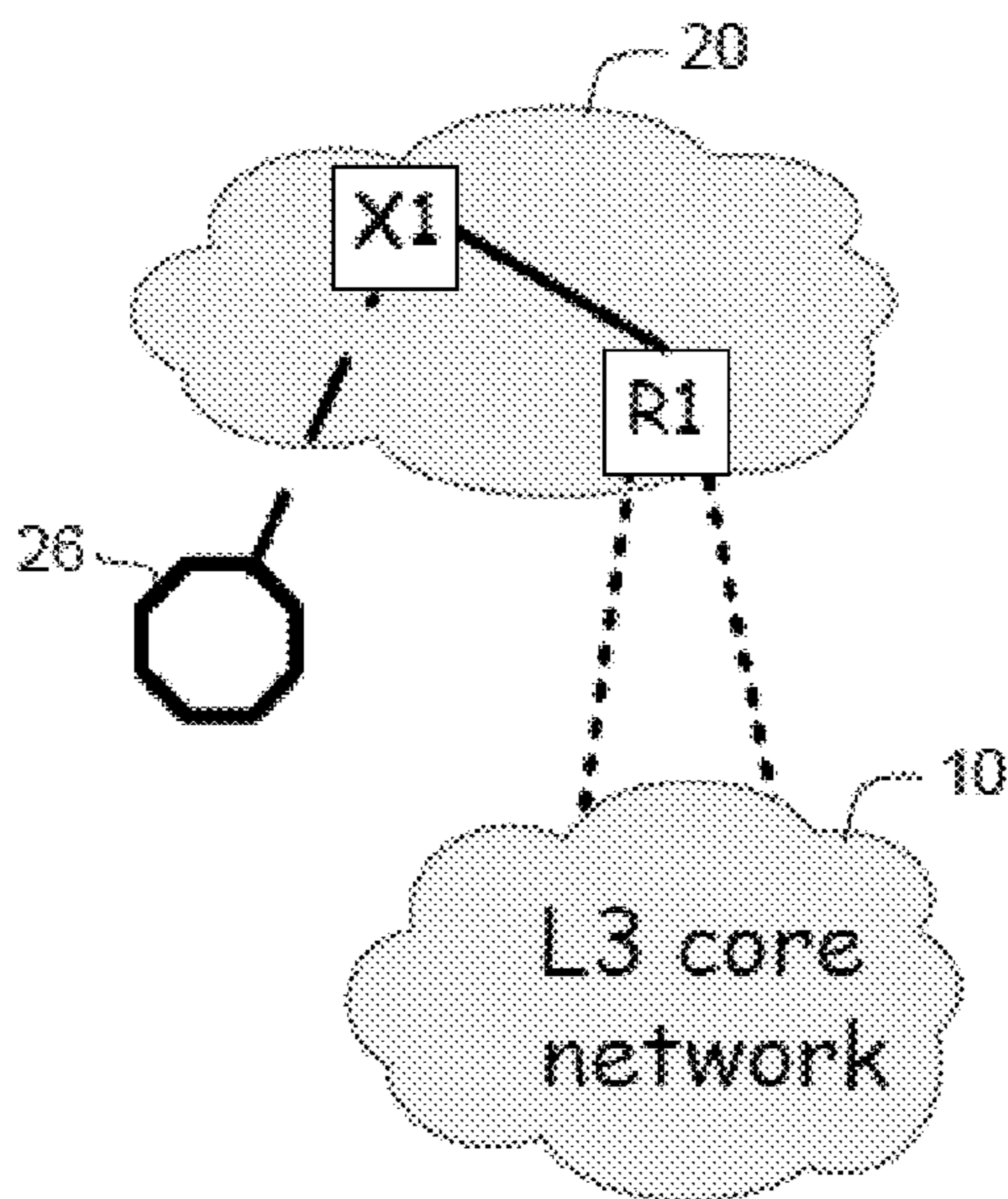


FIGURE 5B

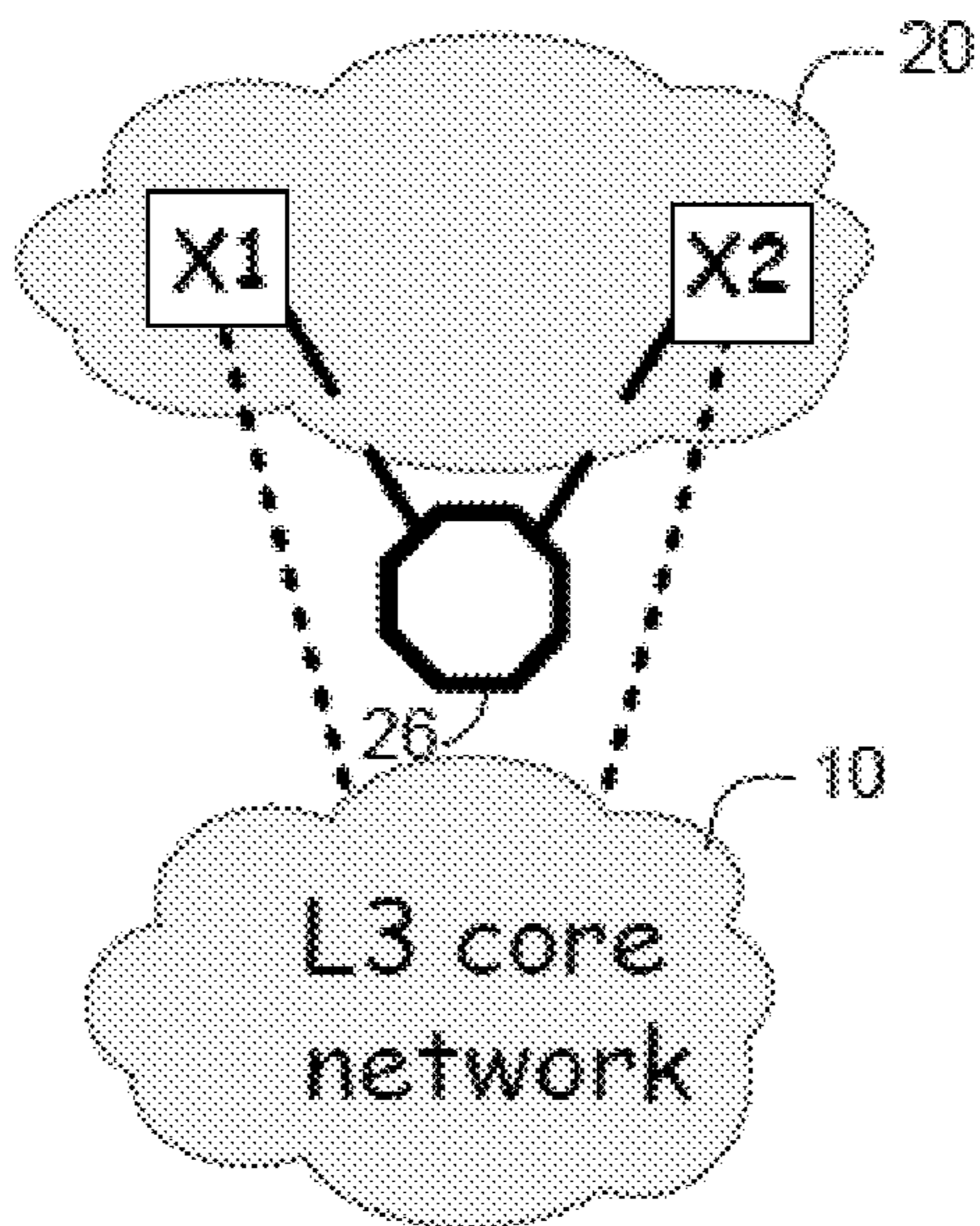


FIGURE 5C

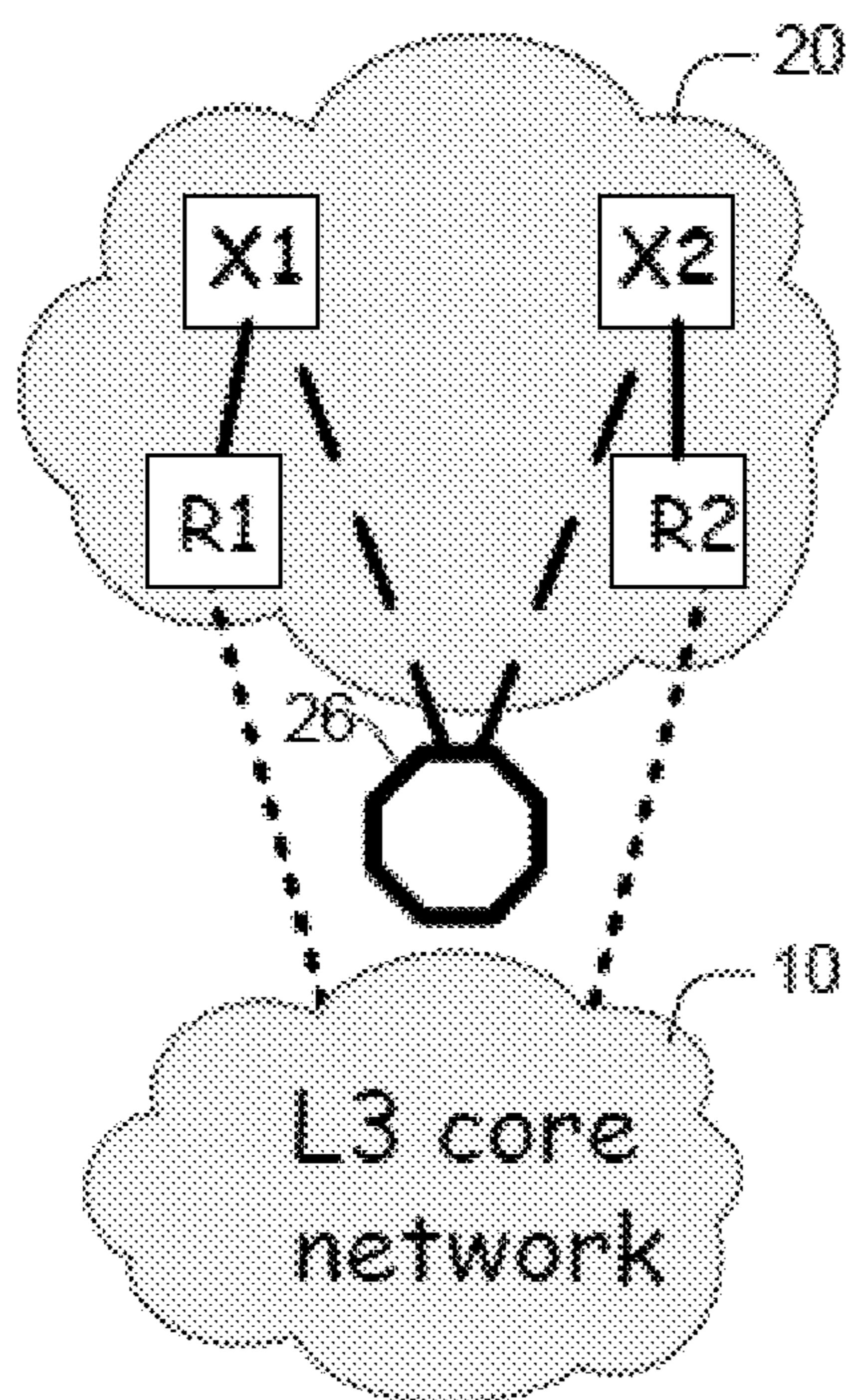


FIGURE 5D

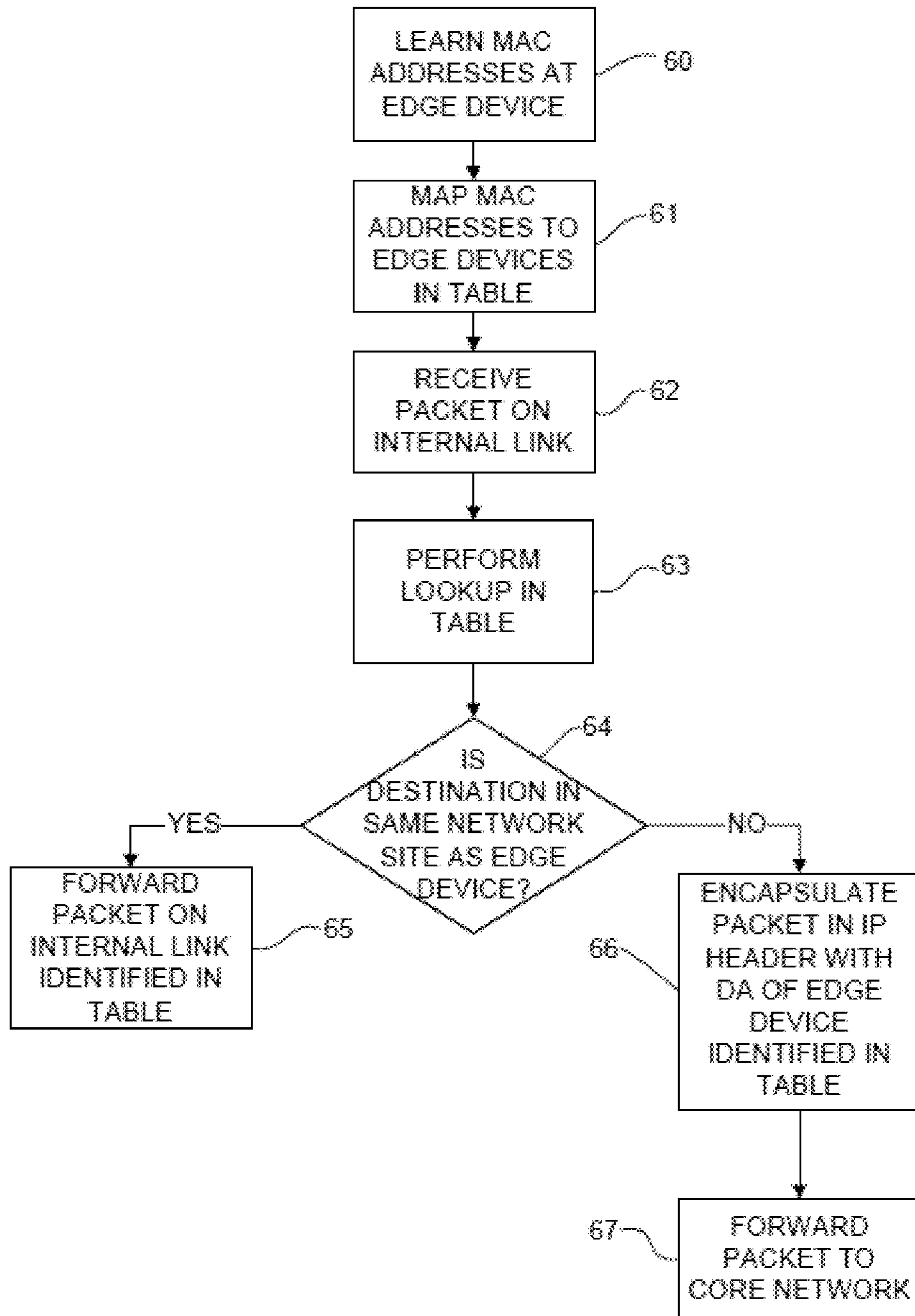


FIGURE 6A

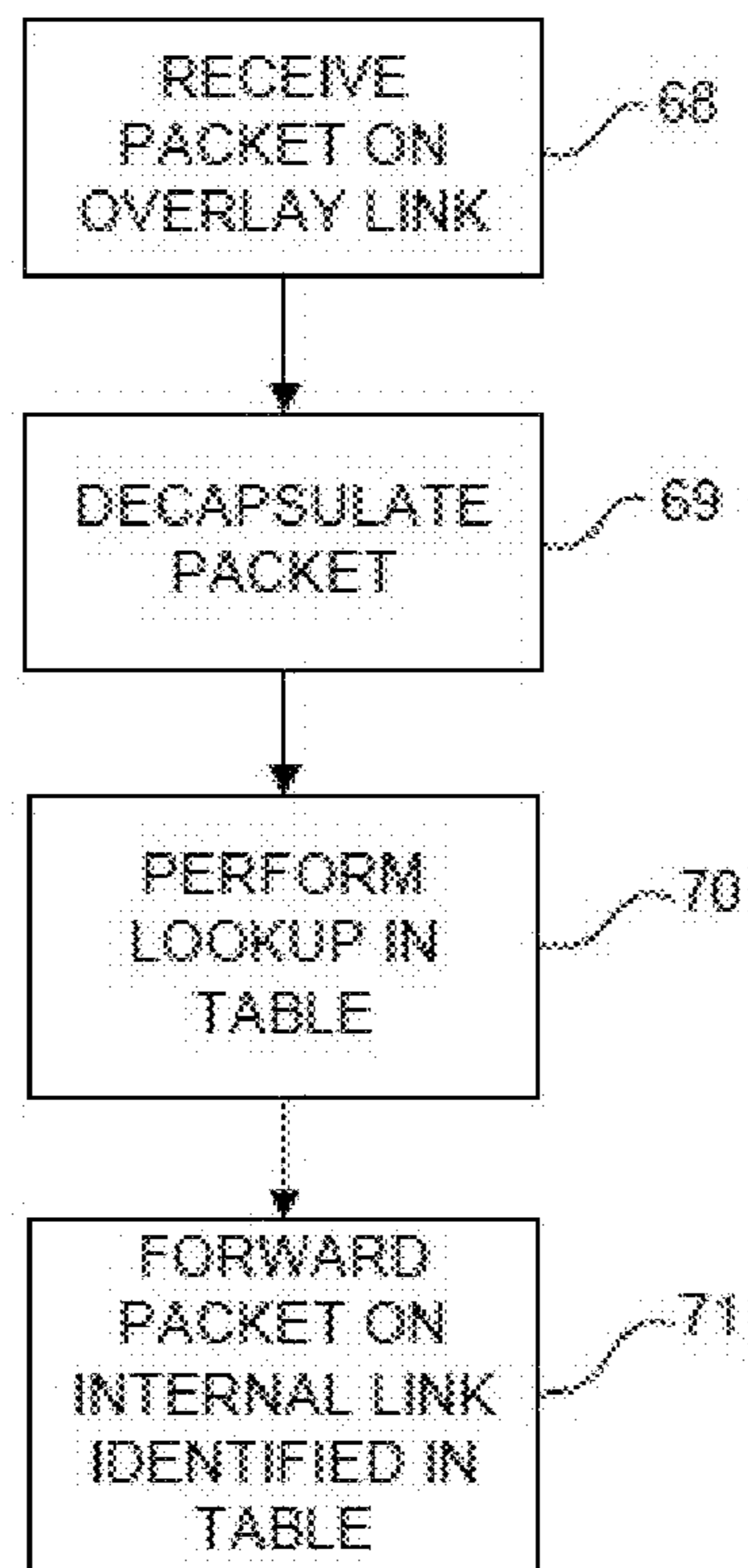


FIGURE 6B

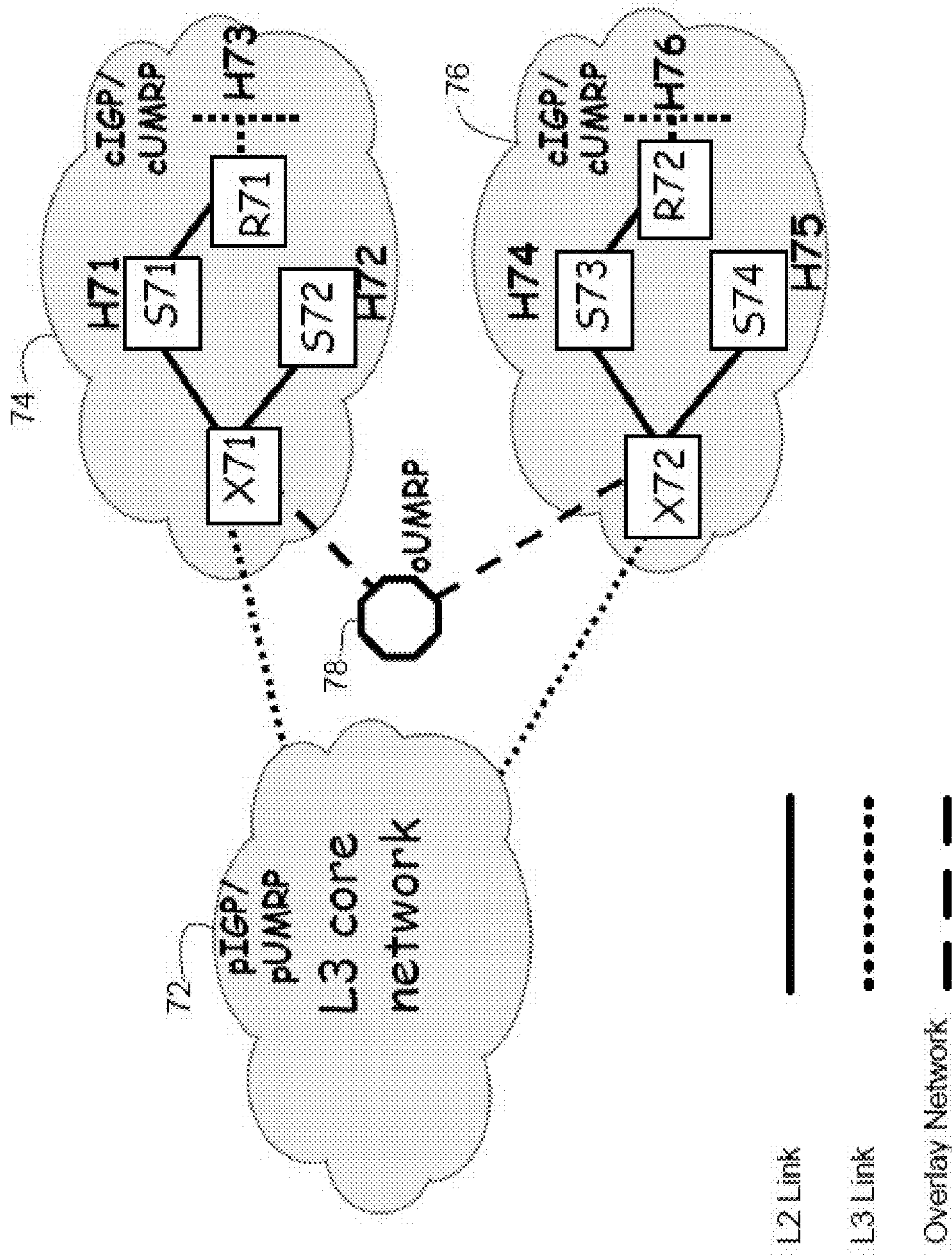


FIGURE 7

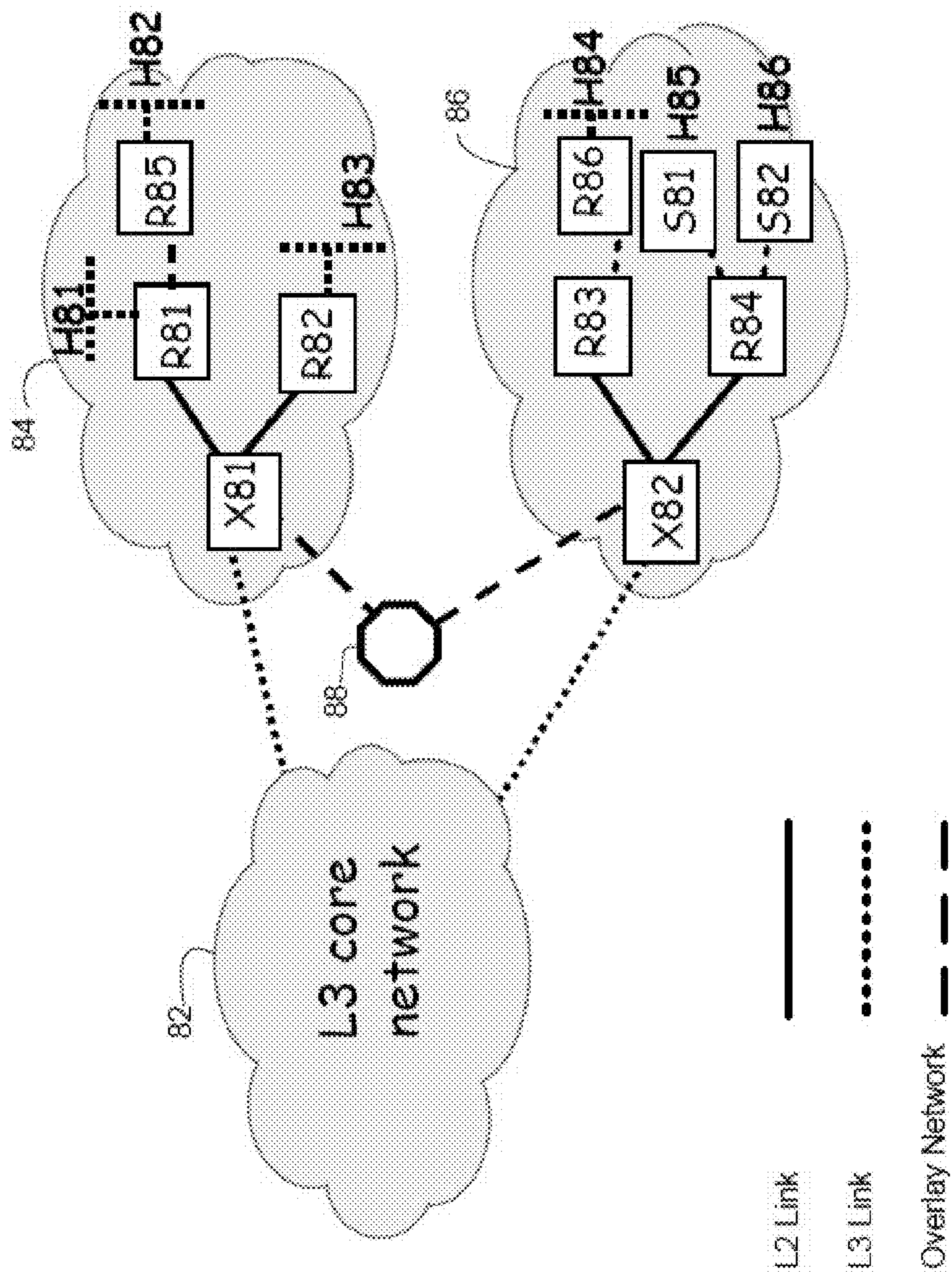


FIGURE 8

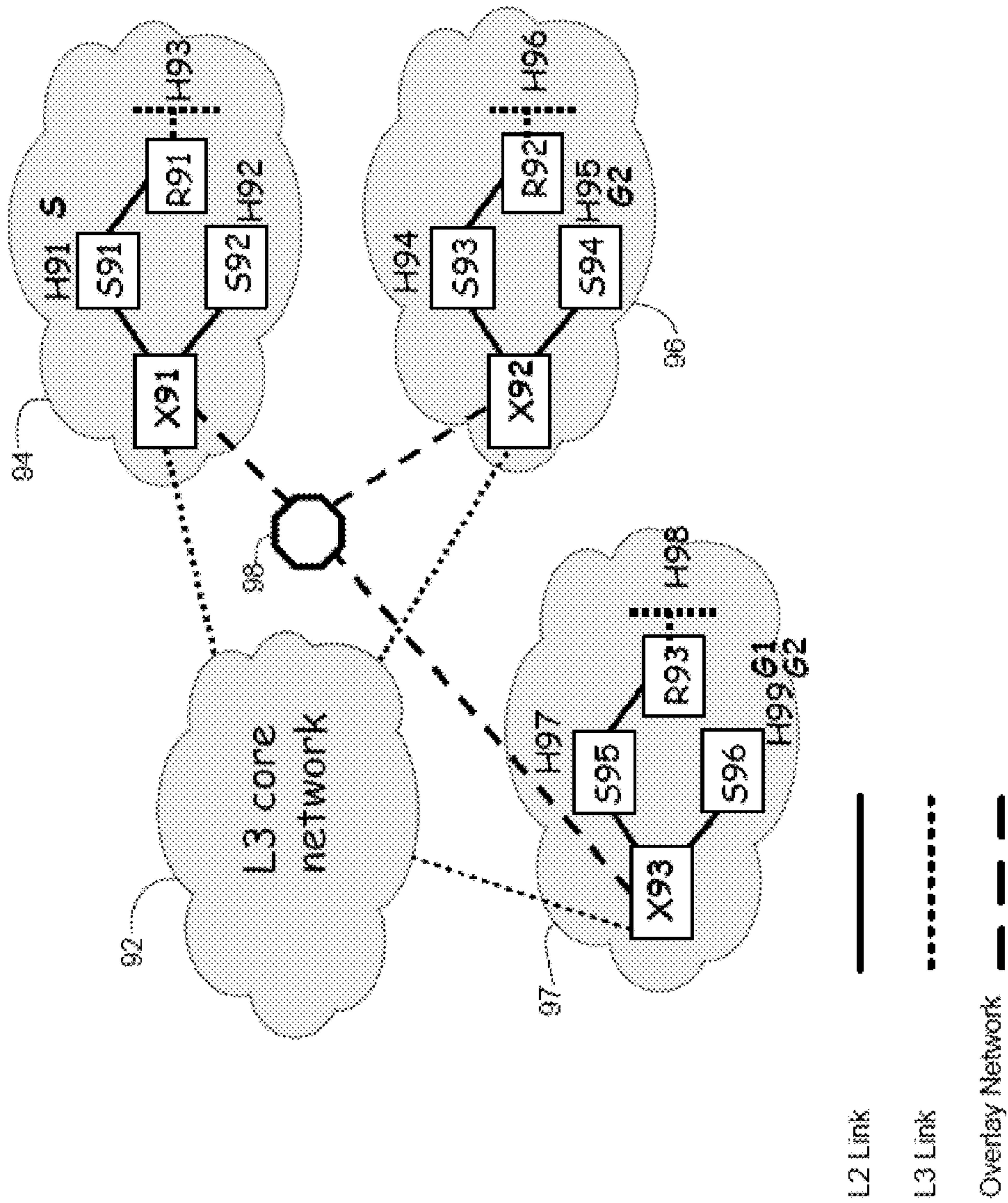


FIGURE 9

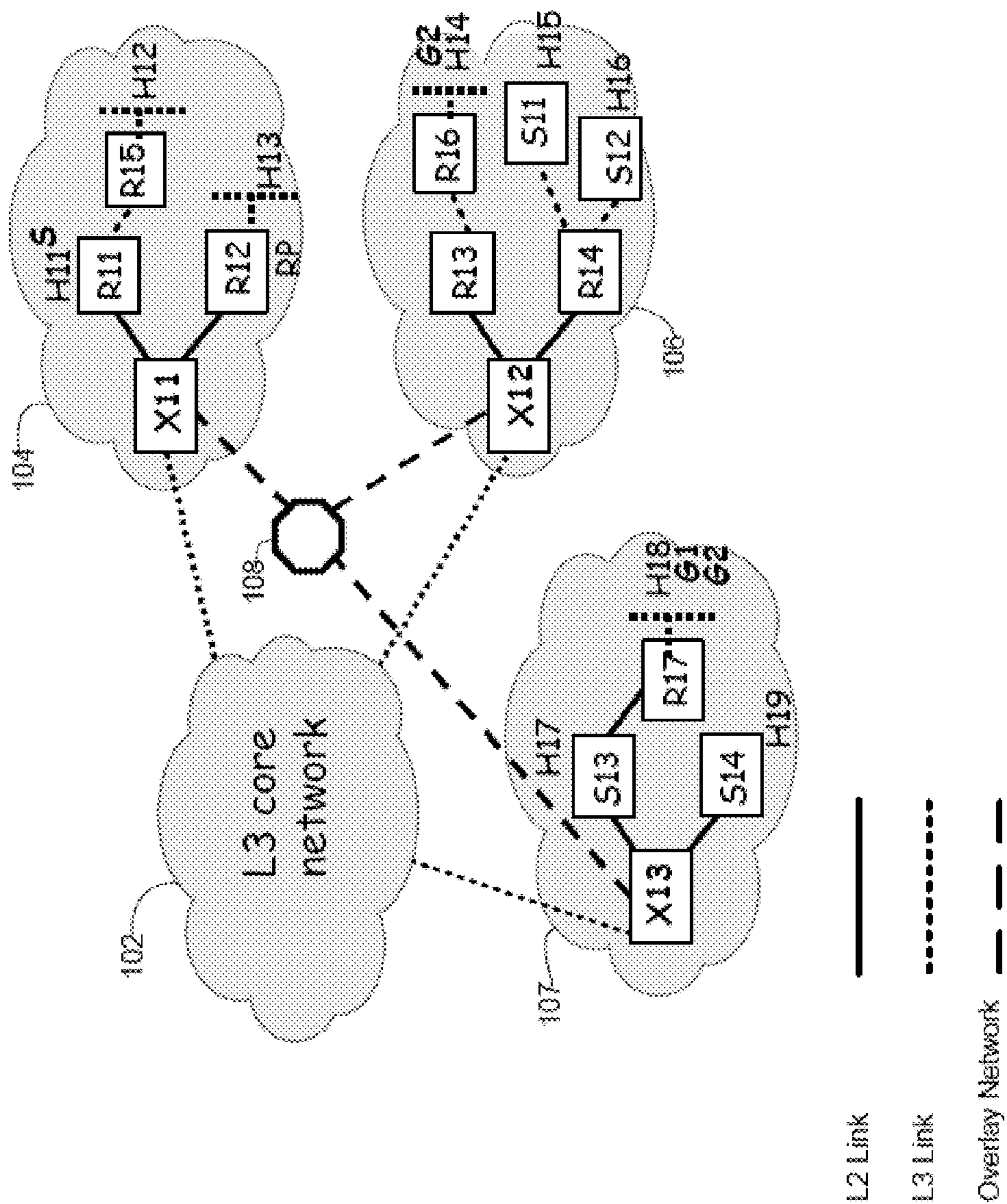


FIGURE 10

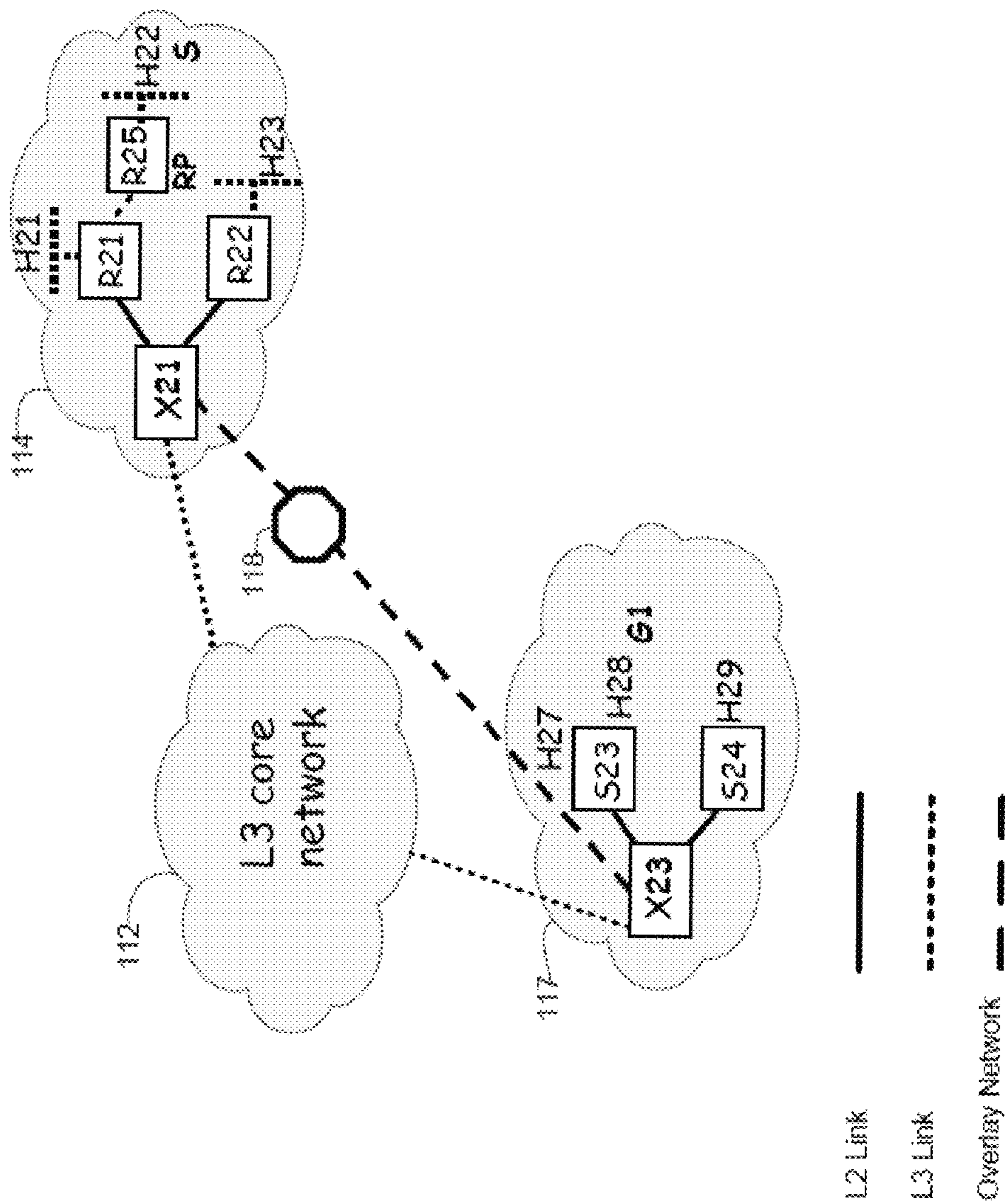


FIGURE 11

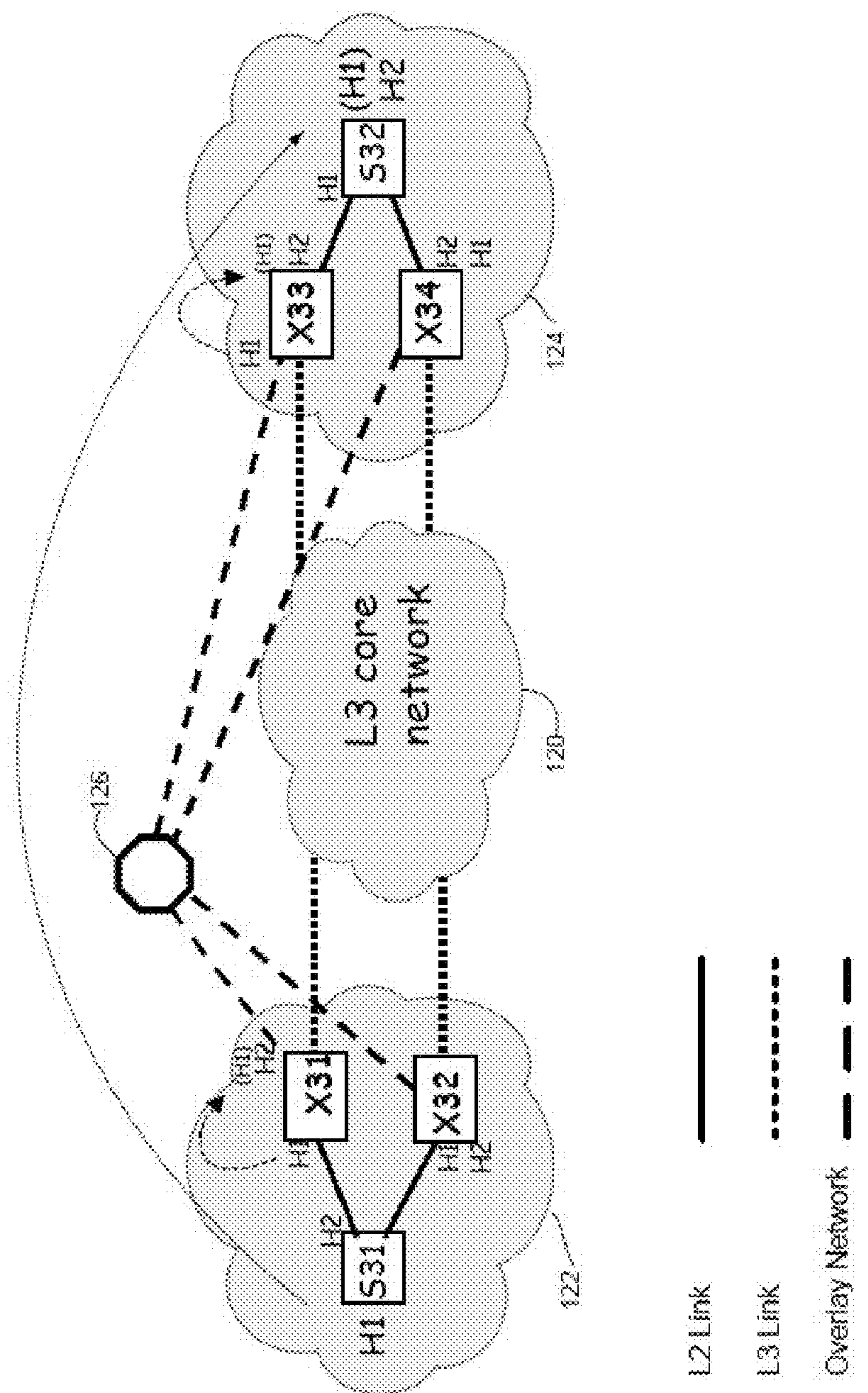


FIGURE 12

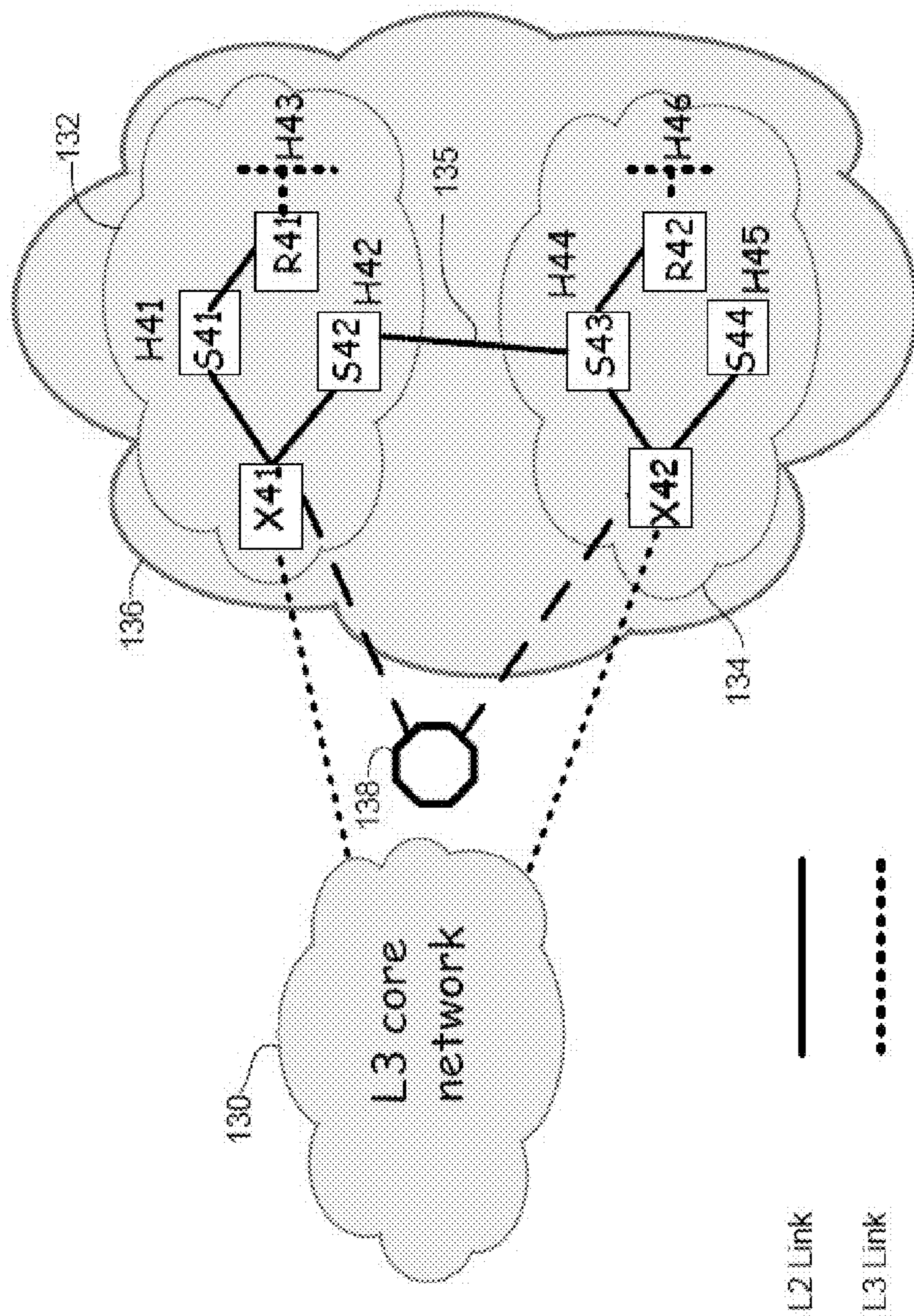


FIGURE 13

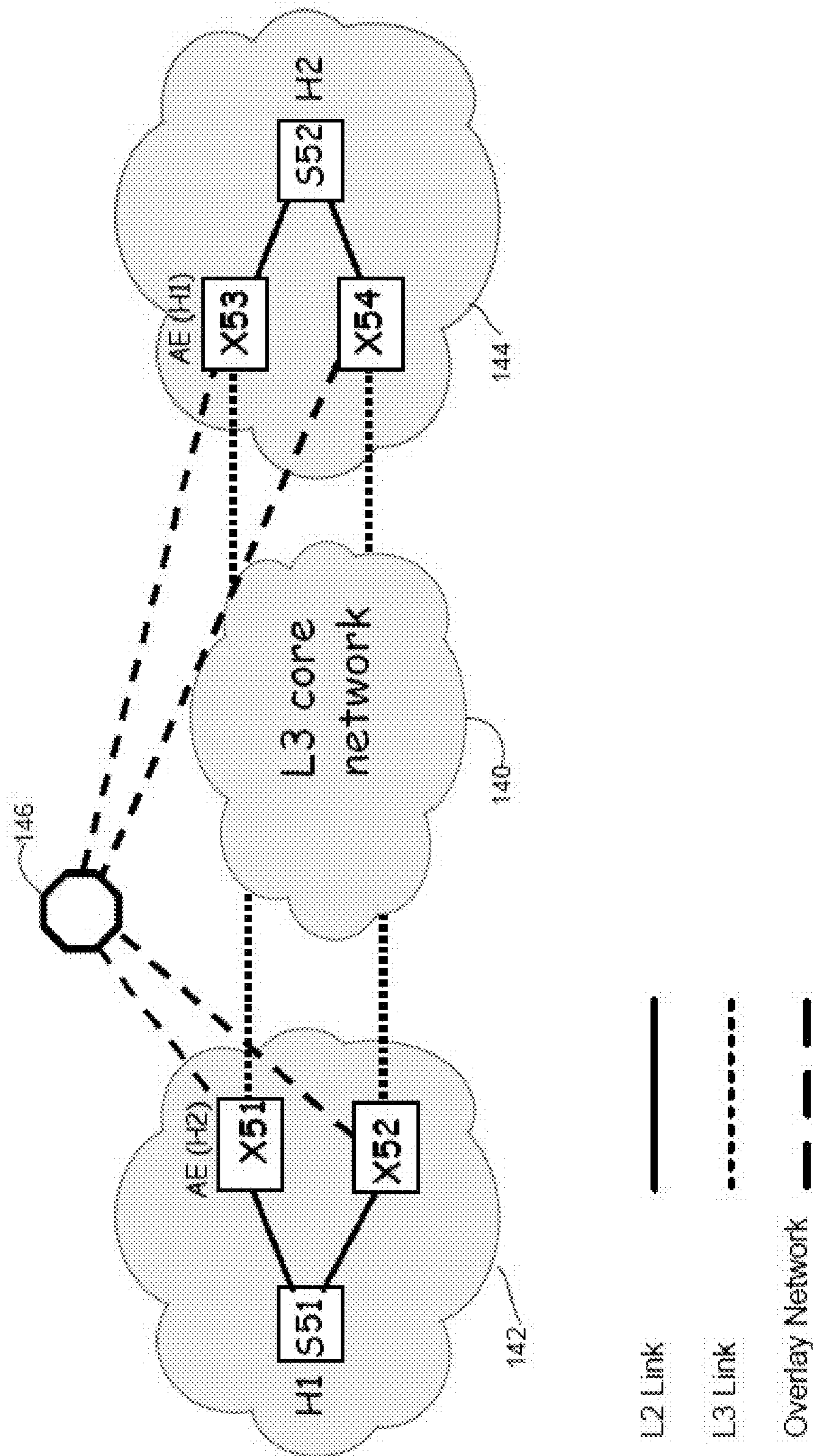


FIGURE 14

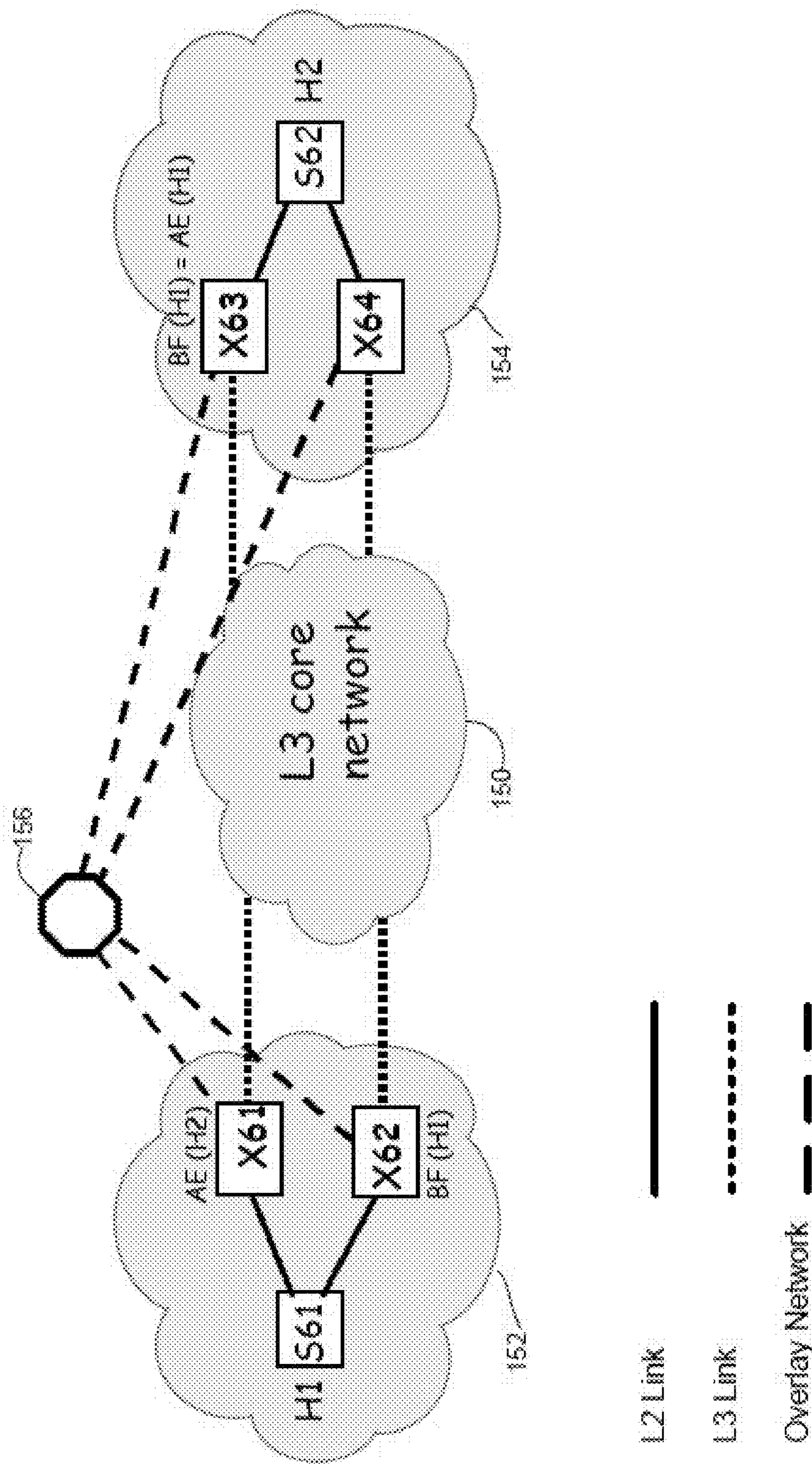


FIGURE 15

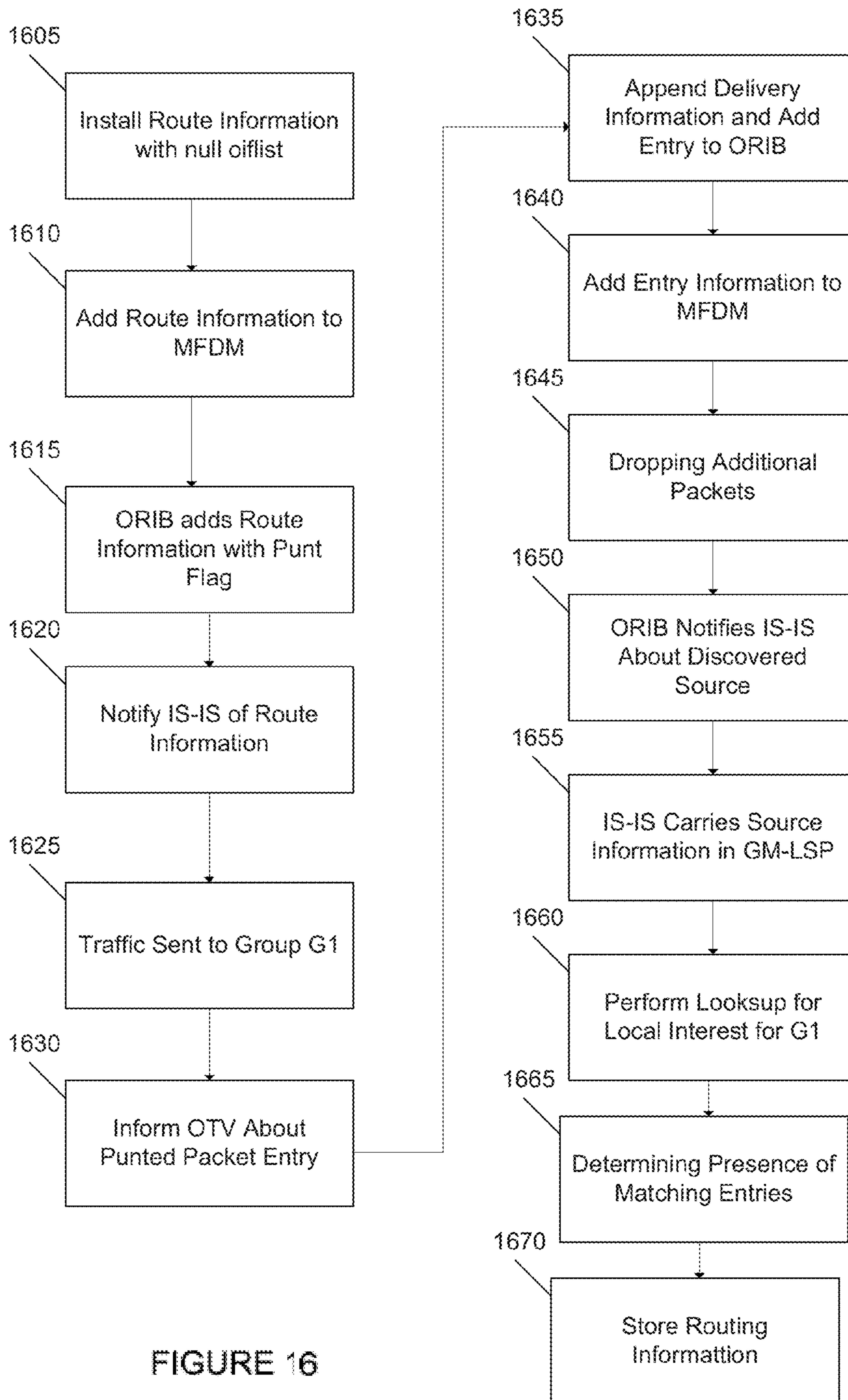


FIGURE 16

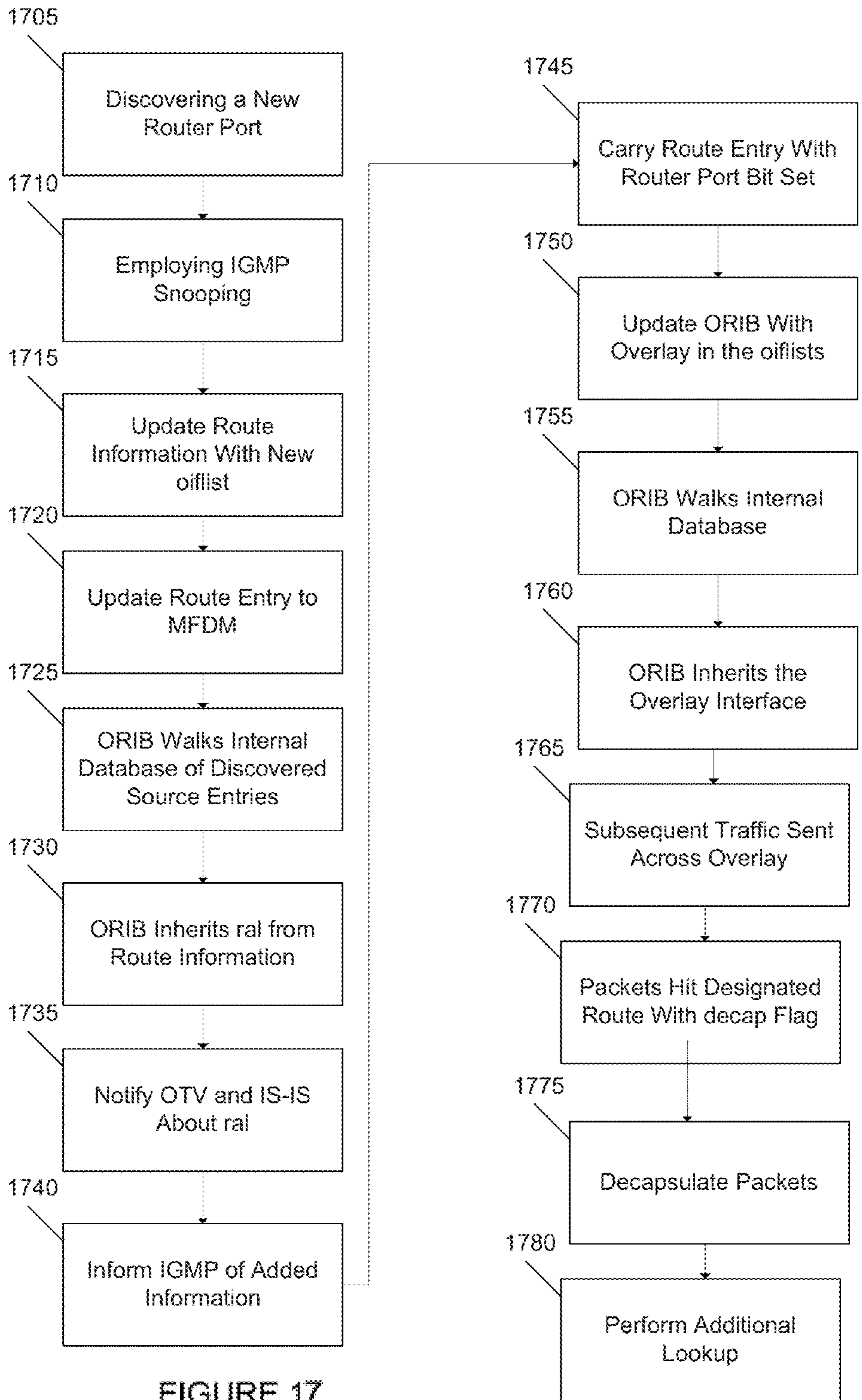


FIGURE 17

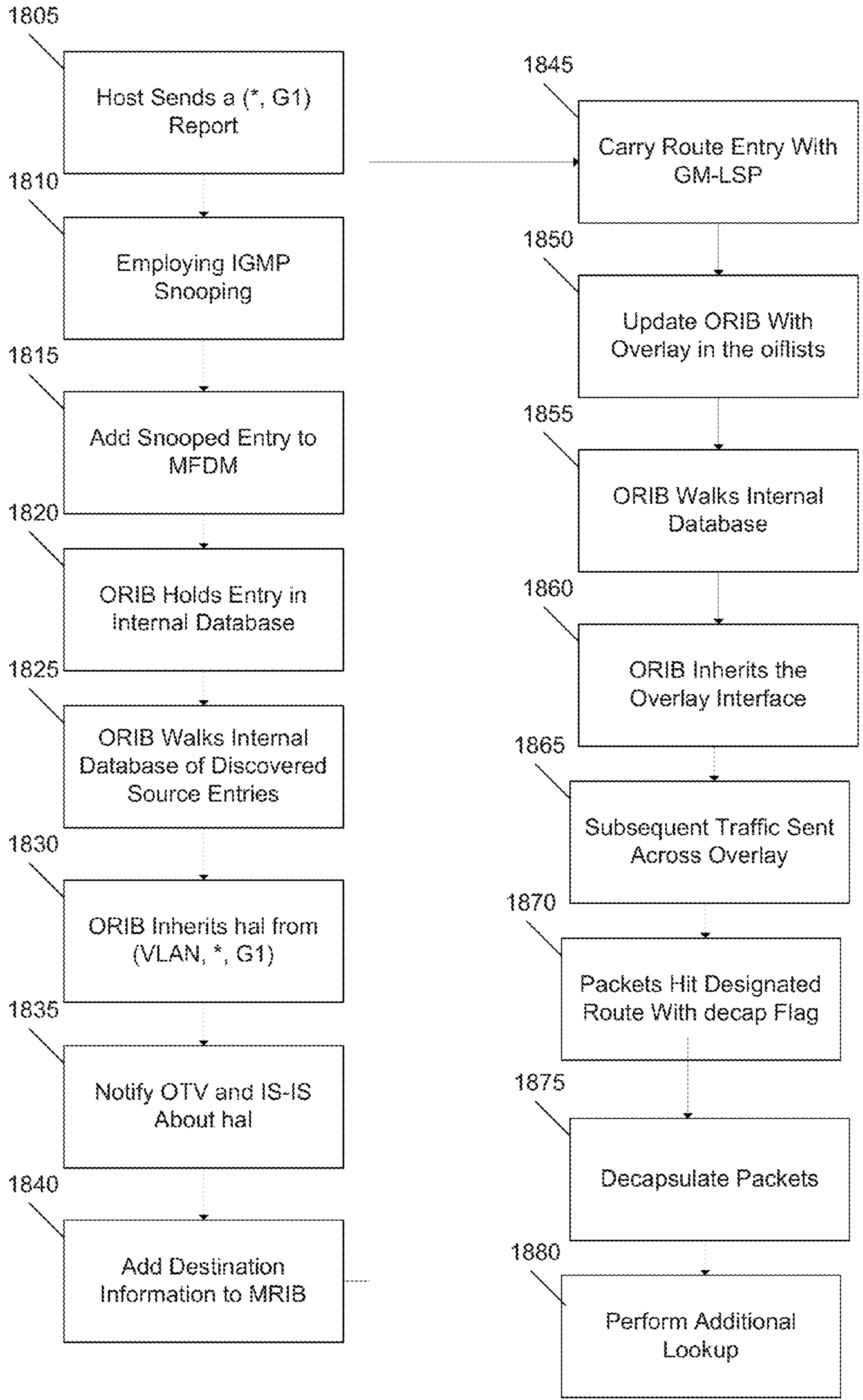


FIGURE 18

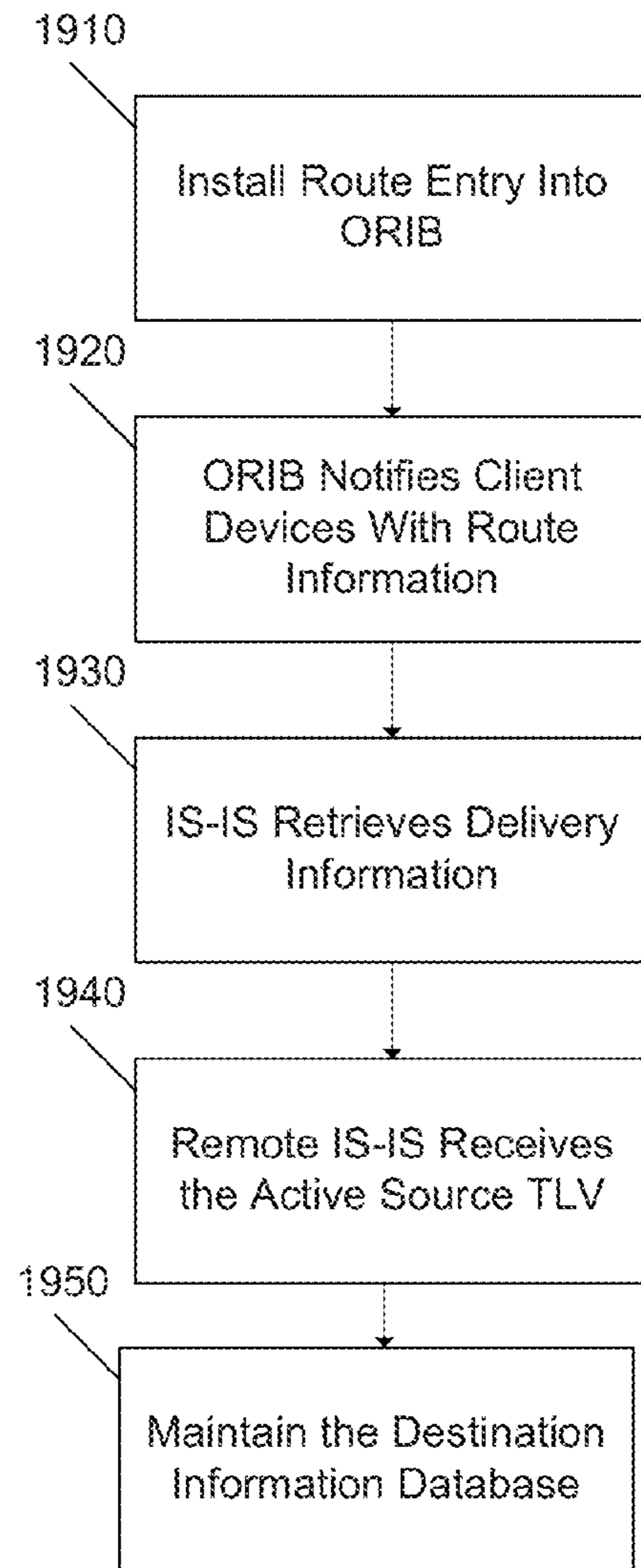


FIGURE 19

1

**MULTICAST ACTIVE SOURCE DISCOVERY
AND MANAGEMENT FOR LAYER-2
INTERCONNECT SOLUTIONS**

TECHNICAL FIELD

The present disclosure relates generally to dynamic discovery and management of multicast active sources in an L2 site.

BACKGROUND

The OTV solution provides for an L2/L3 Virtual Private Network service for both unicast and multicast traffic. However, in order to optimize the multicast state and bandwidth in the service provider core, a mechanism of dynamic discovery of multicast streams is desired. There is a need for a system to enable an OTV edge device in the source-site to dynamically map these streams to service provider multicast groups and announce the mapping to all remote sites to allow the remote receiver sites to trigger the creation of service provider multicast data trees in order to receive the streams.

BRIEF DESCRIPTION OF THE DRAWINGS

Many aspects of the disclosure can be better understood with reference to the following drawings. The components in the drawings are not necessarily to scale. Emphasis is instead placed upon clearly illustrating the principles of the present disclosure. Moreover, in the drawings, like references numerals designate corresponding parts through the several figures.

FIG. 1 illustrates an example of a network in which embodiments described herein may be implemented.

FIG. 2 is a block diagram illustrating additional details of an edge device of the network of FIG. 1, according to one embodiment.

FIG. 3 depicts an example of a network device useful in implementing embodiments described herein.

FIGS. 4A-4E illustrate examples of different edge node placements in a network utilizing embodiments described herein.

FIGS. 5A-5D illustrate different examples of redundant network designs in which the embodiments described herein may be implemented.

FIG. 6A is a flowchart illustrating a process for overlay transport virtualization in which packets are forwarded onto an overlay network, in accordance with one embodiment.

FIG. 6B is a flowchart illustrating a process for overlay transport virtualization in which packets are received from the overlay network, in accordance with one embodiment.

FIG. 7 illustrates an example of unicast packet flow between Layer 2 sites.

FIG. 8 illustrates an example of unicast packet flow between Layer 3 sites.

FIG. 9 illustrates an example of multicast packet flow between Layer 2 sites.

FIG. 10 illustrates an example of multicast packet flow between Layer 3 sites.

FIG. 11 illustrates an example of multicast packet flow in a hybrid VPN.

FIG. 12 illustrates an example of MAC mobility.

FIG. 13 illustrates an example of a network merge.

FIG. 14 illustrates a unicast hash example for use in multi-homing.

2

FIG. 15 illustrates a broadcast hash example for use in multi-homing.

FIG. 16 illustrates a flow chart description of embodiments of the present disclosure.

FIG. 17 illustrates a flow chart description of embodiments of the present disclosure.

FIG. 18 illustrates a flow chart description of embodiments of the present disclosure.

FIG. 19 illustrates a flow chart description of embodiments of the present disclosure.

DETAILED DESCRIPTION OF EXAMPLE
EMBODIMENTS

In various embodiments, a method may be provided comprising installing a route entry associated with multicast traffic to a memory. Client devices may be notified of the route entry for advertisement by an active source device. The delivery group and delivery source may be retrieved from the information for the route entry. Multicast data trees may maintain delivery group and delivery source information for access.

Embodiments of the present invention for multicast active source discovery may be implemented in hardware, software, firmware, or a combination thereof (collectively or individually also referred to herein as logic). To the extent certain embodiments, or portions thereof, are implemented in software or firmware, executable instructions or code for performing one or more tasks of multicast active source discovery are stored in memory or any other suitable computer readable medium and executed by a suitable instruction execution system. In the context of this document, a computer readable medium is an electronic, magnetic, optical, or other physical device or means that can contain or store a computer program for use by or in connection with a computer related system or method.

To the extent embodiments, or portions thereof, are implemented in hardware, the present invention may be implemented with any or a combination of the following technologies: a discrete logic circuit(s) having logic gates for implementing logic functions upon data signals, an application specific integrated circuit (ASIC) having appropriate combinational logic gates, programmable hardware such as a programmable gate array(s) (PGA), a field programmable gate array (FPGA), etc.

In order to efficiently transport multicast traffic between L2 OTV sites across an L3 multicast service provider core, it may be desired to ensure that the traffic is only sent to receiver sites based on the receivers' interest in groups and sources. Furthermore, it may be desired to build provider multicast trees which are rooted at source sites only. This may avoid unnecessary multicast states in the core.

As such, embodiments described in this specification disclose the building of different provider multicast data trees rooted at a source site to receiver site for individual per-vlan (*,G)/(S,G) streams. G represents the group to which the traffic is directed and S refers to the source site from which the traffic originates. Building the provider multicast trees may require individual sites to discover multicast sources local to their network and subsequently announce to all receiver sites the multicast data trees to be used to deliver such traffic. Embodiments to discover such sources are described below.

The present disclosure provides an overview of OTV operations to provide an L2/L3 VPN service for both unicast

and multicast traffic and uses the IS-IS link-state protocol (described below) as the overlay Interior Gateway Protocol (IGP).

Embodiments of the present disclosure may depend on the core provider network having the capability to deliver IP multicast frames originated by any L2 site. Multiple streams originated at a L2 site may be aggregated into one or more multicast delivery trees in the provider core. The mapping of site multicast streams to a delivery multicast tree in the core may be controlled by the site OTV edge device where the traffic is originated.

Overlay Transport Virtualization Overview

A method and system described herein support Layer 2 (L2) and Layer 3 (L3) virtual private networks (VPNs) over a L2 or L3 infrastructure. The method and system are referred to herein as Overlay Transport Virtualization (OTV) since they provide an “overlay” method of doing virtualization versus traditional “in-the-network” type systems, in which multiple routing and forwarding tables are maintained in every device between a source and a destination. With OTV, state is maintained at the network edges, but is not required at other devices in a network site or in a core network. OTV operates at edge devices interposed between the network sites and the core network. The edge devices perform L2 learning and forwarding functions (similar to a traditional L2 switch) on their site-facing interfaces (internal interfaces) and perform IP based virtualization functions on their core-facing interfaces, for which an overlay network is created. The dual functionality of the edge device provides the ability to connect L2 networks, L3 networks, or hybrid (L2 and L3) networks together. OTV can also run at the same time as the network sites and the core network are operating “in-the-network” virtualization via VRF (Virtual Routing and Forwarding) or VNET (Virtual Network).

OTV encapsulates L2 traffic with an IP header (“MAC in IP”), rather than creating stateful tunnels. L2 traffic that requires traversing the overlay network to reach its destination is prepended with an IP header which ensures the packet is delivered to the edge device that provides connectivity to the L2 destination in the original MAC (Media Access Control) destination header. Traffic is forwarded natively in the core network over which OTV is deployed, based on the IP header. The native IP treatment of the encapsulated packet allows optimal multi-point connectivity as well as optimal broadcast and multicast forwarding. OTV is independent of the technology deployed in the core network and no changes are required to core devices.

As described in detail below, OTV provides a state to map a given destination MAC address in a L2 VPN to the IP address of the OTV edge device behind which that MAC address is located. OTV forwarding is therefore a function of mapping a destination MAC address in the VPN site to an edge device IP address in the overlay network. A control plane is used to exchange reachability information among the different OTV edge devices. In one embodiment, the control plane uses an overlay IGP (Interior Gateway Protocol) to carry MAC addresses and IP addresses. The MAC addresses are those of hosts connected to the network and the IP next hops are the addresses of the edge devices through which the hosts are reachable through the core.

In one embodiment, the control plane utilizes IS-IS (Intermediate System-to-Intermediate System) as an IGP capable of carrying a mix of MAC unicast and multicast addresses as well as IP addresses. The information carried in IS-IS LSPs (Link State Packets) is MAC unicast and multicast addresses with their associated VLAN IDs (or VNIDs in the case of L3 VPNs) and IP next hops. Since all MAC

addresses on a site are advertised in IS-IS to all other sites, all edge devices have knowledge of all MAC addresses for each VLAN in the VPN. It is to be understood that routing protocols other than IS-IS may be used, without departing from the scope of the invention.

OTV Network

Referring now to the drawings, and first to FIG. 1, an example of a network in which OTV may be implemented is shown. Some of the elements in a network that employs OTV may be routers, switches, gateways, servers, or other network devices.

The network shown in FIG. 1 comprises a virtual private network (VPN A) including three network sites; site 1 (14), site 2 (16), and site 3 (18) and VPN B including three network sites; site 4 (20), site 5 (22), site 6 (24). Each VPN is a collection of sites which are typically controlled by a single administration or organization. The addressing plan, router, and switch configuration is consistent as it would be if the sites were physically at the same location. The sites may be a single or multi-homed connected network. The network sites are in communication with other network sites through a core network 10. The core network 10 may be a L2 metro Ethernet core, L3 IP network core, or an MPLS core, for example. As previously noted, OTV is independent of the technology deployed in the core network. OTV may be implemented, for example, on any network capable of correctly forwarding IP packets.

Site 4, site 5, and site 6 each include an edge device 12, which is configured to run OTV. OTV defines an overlay network 26 coupled to each edge device 12 at the customer sites 20, 22, 24. The sites are thus connected together via the edge devices 12 that operate in the overlay network 26 and provide Layer 2 and Layer 3 connectivity among the sites.

In the example shown in FIG. 1, there is one overlay network per VPN. For 20 simplification only one overlay network 26 is shown for VPN B. VPN A also has an overlay network coupled to edge devices of site 1, site 2, and site 3 (not shown). There may therefore be multiple overlay networks operating within a network. In the embodiments described herein, there is a single IS-IS process for each overlay network 26. However, there may also be multiple IS-IS processes running within an overlay network or on a single edge device, with each serving a different VPN. IS-IS may also be configured so that it is fault tolerant across VDCs (Virtual Device Contexts).

The network sites 20, 22, 24 may be L2 sites, L3 sites, or a combination thereof. Examples of L2 and L3 networks are described below and shown in FIGS. 7 and 8, respectively. Each client network 20, 22, 24 may run Spanning Tree. In the case where Spanning Tree is used by the client, each client network 20, 22, 24 has its own client Spanning Tree Protocol (cSTP) and its own STP root bridge. STP is not extended on the overlay network 26 since there is no need to create one large STP domain across sites. Each client network also performs its own snooping (Snoop (i)) (e.g., IGMP/MLD (Internet Group Management Protocol/Multicast Listener Discovery) snooping) at L2. IGMP/MLD reports originated at an OTV site are not forwarded on the overlay network 26.

Since OTV can work over an L2 core transport or an L3 core transport, a transport independent mechanism is provided to allow peering between edge devices 12. The edge devices include an IP address on their core facing interface and preferably join a configured ASM (Any Source Multicast)/Bidir multicast group in the core transport network 10 by sending IGMP reports. The edge devices 12 are therefore hosts relative to the core, subscribing to multicast groups

that are created in the provider network and which rely on a provider Unicast/Multicast Routing Protocol (pUMRP).

In order for the OTV edge devices **12** to provide transparency to any bridged domains connected thereto and eliminate unnecessary flooding, the edge devices combine data plane learning on their bridged internal interfaces with control plane learning on their overlay interfaces. The triggering of control plane advertisements and learning by data plane events, along with the interaction of the overlay control plane with the provider control plane and the customer control plane is described below.

In one embodiment, data-plane hashing may be used so that different MAC addresses from the same VLAN can use different edge devices in an active-active capable matter.

In one embodiment, the OTV network is configured to provide encryption by the core network **10**, client networks **20**, **22**, **24**, or edge devices **12**. Edge devices **12** may maintain security associations directly amongst themselves and may use an L2 and L3 encryption scheme. For example, the edge devices may provide IPsec (IP Security) encryption or Link-sec encryption.

Control Planes

OTV leverages three control planes which support each other while operating independently. A provider control plane enables unicast reachability among the edge devices **12** and also provides the multicast group that makes the edge devices adjacent from the overlay control plane perspective. An overlay control plane is used to exchange reachability information among different OTV edge devices **12**. A client control plane may run across the overlay control plane to get the reachability necessary between client routers to establish routing adjacency in the case of L3 VPNs.

As noted above, the control planes operate independently, however, in order to optimize multicasting, multicast control plane events (e.g., reports, joins, leaves) that occur in one MRP may initiate events in another MRP so that the optimal tree is always being used to forward traffic. Also, events in the overlay control plane may be triggered by forwarding events in the client data plane.

The provider control plane includes the set of routing protocols which run in the core infrastructure to be able to deliver packets routed from the site networks **20**, **22**, **24**. The edge devices **12** are allocated an IP address out of the core block of addresses. The multicast group that the edge devices join is referred to herein as the Provider Multicast Group (pMG). The pMG is used for edge devices to become adjacent with each other to exchange, for example, IS-IS LSPs, CSNPs, and Hellos. Thus, by virtue of the pMG, all edge devices see each other as if they were directly connected to the same multi-access multicast-capable segment for the purposes of IS-IS peering. In one embodiment, each VPN uses a unique IS-IS authentication key and a dedicated ASM/Bidir multicast group **20** (pMG) allocated by the core network. The pMG also defines a VPN, therefore, when an edge device joins a pMG, the site becomes part of a VPN. Multiple pMGs define multiple VPNs. The pMG may also be used to broadcast data traffic to all edge devices in the VPN when necessary.

In an alternative embodiment, a full mesh of tunnels can be substituted for the pMG as a way of connecting the edge devices for a given VPN in the overlay network. The configuration of these tunnels may be performed manually or through the use of a discovery mechanism. Head-end replication at the edge devices would be provided for control-plane packets and data-plane packets that need to reach all edge devices in the VPN.

The overlay control plane is referred to herein as overlay Unicast/Multicast Routing Protocol (oUMRP). In order to avoid flooding of unknown unicast packets among edge devices, OTV does not perform data-plane learning on the overlay network. Data-plane learning takes place on the internal interfaces to provide compatibility and transparency within the L2 sites connecting the OTV overlay network **26**. Regardless of whether OTV is supporting L2 or L3 VPNs, the edge devices **12** appear to each VPN site to be providing L2-switched network connectivity amongst those sites.

The L2-switched overlay network **26** transparently transports the control and data plane traffic for the routed network. The routed or switched customer network is referred to herein as the client network, and cIGP (client IGP) and cUMRP (client Unicast/Multicast Routing Protocol) are used to identify the client control plane components.

MAC learning across the overlay network **26** is based on advertisement of unicast and multicast MAC addresses, which is done by the overlay control plane. The (sitefacing) MAC addresses that are locally connected to an edge device **12** are advertised in the overlay IGP to other edge devices in the VPN. The overlay IGP establishes adjacencies only between edge devices that are in the same VPN. Edge devices become part of a VPN when they join a multicast group defined in the core (provider-MRP). The Hellos and updates between overlay-IGP peers travel over the multicast group defined in the pMRP. Thus, edge devices peer with each other as if they were directly connected at L2. This peering is possible as all the traffic for the oUMRP is encapsulated with the pMRP group address and sent into the core. Thus, all edge devices in a given VPN receive the oUMRP multicast traffic as if they were all on the same segment.

The oUMRP control traffic is encapsulated with the pMRP group address corresponding to the VPN. The oUMRP is used to inform all edge devices that the subscribers to a particular group are reachable over the overlay network. In one embodiment, the edge devices **12** snoop IGMP/MLD reports and then the oUMRP notifies all edge devices in the VPN which group has been joined by sending a GM-LSP (Group Membership-Link State Packet) with the group MAC address in it. The IP multicast group address is also sent. The GM-LSP is used to store and advertise multicast MAC (mMAC) addresses learned at a site **20**, **22**, **24**. The information conveyed by the oUMRP is used by the edge devices **12** to populate their oif list at the source site. Edge devices **12** on the receiving sites can join the corresponding (S, G) group in the provider plane (pMRP) when they snoop the IGMP/MLD traffic from the site. Thus, multicast trees are built natively in the core network **10**, and not in the overlay network **26**.

The edge device **12** does not participate in the provider IGP as a router, but as a host. As previously discussed, the edge device **12** has an IP address which is significant in the core/provider addressing space and joins multicast groups in the core network **10** by issuing IGMP/MLD reports. Thus, the edge device does not have an IGP relationship with the core, nor does it have any PIM adjacencies with the core. However, the edge device **12** does participate in the oUMRP and its IP address is used as a router ID and a next hop address for unicast traffic by the oUMRP. The edge device **12** does not build an IP routing table with the information received from the overlay IGP, but instead builds a hybrid table (described below) where MAC address destinations are reachable via IP next hop addresses. The edge devices **12** thus operate as IP hosts in the provider plane, MAC routers in the overlay plane, and bridges in the client bridging plane.

The following describes details of the edge device and OTV operation at the edge device, including address advertisement, forwarding, and encapsulation.

OTV Edge Device

FIG. 2 illustrates additional details of an edge device 12, according to one embodiment. The term “edge device” as used herein refers to an L2 device that performs OTV functions. The edge device 12 will typically run as a L2 device (e.g., switch) but can be co-located in a device that performs L3 routing on other L3-enabled ports. For example, the device may perform MAC learning (i.e., site-facing) to learn the MAC addresses of attached routers and run cIGP to learn about host reachability. It is to be understood that the references herein to an L2 switch include devices that perform L3 routing.

The edge device 12 may include, for example, a master central processing unit (CPU), interfaces, and a bus (e.g., a PCI bus). When acting under the control of appropriate software or firmware, the CPU is responsible for such tasks as forwarding table computations, network management, and general processing of packets. It preferably accomplishes all of these functions under the control of software including an operating system and any appropriate applications software. In one embodiment, the edge device 12 is implemented on a general purpose network machine as described below with respect to FIG. 3.

FIG. 3 depicts a network device 50 that may be used to implement embodiments described herein. In one embodiment, network device 50 is a programmable machine that may be implemented in hardware, software, or any combination thereof. A processor 52 executes codes stored in a program memory 54. Program memory 54 is one example of a computer-readable medium. Program memory 54 can be a volatile memory.

Another form of computer-readable medium storing the same codes would be some type of non-volatile storage such as floppy disks, CD-ROMs, DVD-ROMs, hard disks, flash memory, etc. A carrier wave that carries the code across the network is an example of a transmission medium.

Network device 50 interfaces with physical media via a plurality of linecards 56. Linecards 56 may incorporate Ethernet interfaces, DSL interfaces, Gigabit Ethernet interfaces, 10-Gigabit Ethernet interfaces, SONET interfaces, etc. As packets are received, processed, and forwarded by network device 50, they may be stored in a packet memory 58. To implement functionality according to the system, linecards 56 may incorporate processing and memory resources similar to those discussed above in connection with the network device as a whole. It is to be understood that the network device shown in FIG. 3 is only one example, and that network devices having other components or configurations may be used to implement OTV, without departing from the scope of the invention.

Referring again to the example shown in FIG. 2, edge device 12 includes one or more internal interfaces 34, 36 in communication with network devices at the same network site as the edge device, and an overlay interface 32 in communication with the overlay network 26. The internal interfaces 34, 36 are L2 interfaces connected to sitebased switches or site-based routers and learn the source MAC addresses of traffic they receive. The internal interfaces 34, 36 are L2 interfaces regardless if they connect to a switch or a router. Traffic received on an internal interface may trigger oUMRP advertisements or pMRP group joins, as previously described. Traffic received on the internal interfaces 34, 36 is forwarded according to a table 30 onto another internal

interface (regular bridging (or routing in the case of L3 edge device)) or onto the overlay interface (OTV forwarding).

In one embodiment, the forwarding table 30 is a MAC table of 48-bit MAC addresses. The table 30 may contain unicast MAC addresses or multicast MAC addresses. The table 30 is populated by conventional data-plane learning on internal interfaces 34, 36 and by the IS-IS protocol (or other IGP protocol) at the control plane on the overlay interface 32. Arrow 39 illustrates data-plane learning on internal interface 34 and arrow 43 illustrates control-plane learning on the overlay interface 32. The table 30 allows the same MAC address to be used in different VLANs and potentially in different VPNs. The table 30 is described further below with respect to forwarding. The overlay interface 32 is a logical multi-access multicast-capable interface and is realized by one or more physical core facing interfaces. The core facing interface 32 is assigned an IP address (e.g., IPB in FIG. 2) out of the provider/core address space. Even though the overlay interface 32 has an IP address, it does not participate in the provider IGP or UMRP, as noted above. The overlay interface 32 operates as a host connected to the core network 10. The IP address assigned to the overlay interface 32 is used as a next hop address by the oUMRP, therefore, table 30 for the overlay interface includes a remote IP address as the next hop information for remote MAC addresses. As described below, traffic transmitted out of the overlay interface 32 is encapsulated with an IP header and traffic received on this interface is decapsulated to produce an L2 frame.

In the example shown in FIG. 2, there is only one overlay logical interface 32. It is to be understood, however, that there may be any number of overlay interfaces. A processor 28 encapsulates L2 (MAC level) packets 38 from the internal interfaces 34, 36 in IP packets (with unicast or multicast headers) 40 before forwarding the packets to the core network, and decapsulates IP packets received from the core network. For example, in one embodiment a unicast frame is encapsulated in an IP unicast packet; a broadcast frame is encapsulated in an ASM/Bidir IP multicast packet; a link-local multicast frame is encapsulated in an ASM/Bidir IP multicast packet; and a customer multicast frame is encapsulated in a SSM IP multicast packet. The ASM/Bidir multicast packet is also used for IS-IS peering.

In the embodiment utilizing IS-IS, the protocol is operational on the logical interface 32 at the overlay network and, in a constrained fashion, on the internal site network interfaces 34, 36. On the overlay network, IS-IS packets are encapsulated in IP multicast packets. The full IS-IS protocol runs on the overlay link 32. The IS-IS process interacts with the L2 RIB (Routing Information Base) for adding and deleting entries (VLAN-id, MAC). On the site link 34, 36, IS-IS sends IIHs (IS-IS Hellos) for the purpose of detecting reachability inside the site and to elect a site-id. Each edge device which resides in an OTV site advertises over the overlay network the same site-ID. Site-ID election is dynamically determined by the IS-IS protocol. No flooding of LSPs or sending of CSNPs (Complete Sequence Number PDUs) are required, and no pseudo-node LSP needs to be generated for the site links 34, 36.

An edge device 12 that forwards L2 frames in and out of a site from and to the overlay interface, respectively, is an authoritative edge device. There is one authoritative edge device for all MAC unicast and multicast addresses per VLAN. For other VLANs, another edge device is authoritative. The authoritative edge device has the responsibility for forwarding frames from the site network to the overlay network to one or more remote sites, as well as from the

overlay network into the site network. The authoritative responsibility can be shared with other edge devices for other VLANs so that the traffic load can be balanced across edge devices for all VLANs. Therefore, within a VLAN OTV employs an active-backup procedure and across VLANs an active-active procedure.

Authoritative edge device selection may be made, for example, by using static VLAN-ID range assignments. In one embodiment, an edge device ordinal is assigned dynamically by the IS-IS protocol per edge device within a site. The ordinal value ranges from 0 to n-1, where n is the number of edge devices per site. Ordinals are preferably contiguously assigned across all edge devices based on IS-IS system ID. When an edge device goes down, which is detected by an IS-IS adjacency loss on the internal network, the ordinal of the down edge device is taken over ownership by the numerically lower ordinal assigned edge device which is still up.

The authoritative edge device may also be selected by mapping the VLAN-IDs to an edge device ordinal using a hash function. A hash-based scheme ensures a balanced distribution of VLANs across all edge devices, irrespective of the actual values or ranges in use. Another embodiment for selecting authoritative edge devices for load balancing across one VLAN is described below under load balancing and ECMP. It is to be understood that the selection methods described herein are only examples, and that other methods may be used to select the authoritative device.

Advertising Addresses

As previously discussed, in addition to conveying MAC address information among the edge devices in a VPN, information mapping the MAC addresses to IP addresses of an advertising edge device is conveyed for the purpose of "MAC-in-IP" forwarding across the overlay network.

When an edge device is authoritative, it advertises a unicast MAC address as soon as it learns the MAC on an internal interface. The MAC address is placed in an IS-IS LSP if the edge device is authoritative for the VLAN the MAC resides in. It should be noted that the role of being authoritative has no effect on multicast MAC addresses.

Forwarding

The following describes details of forwarding between internal links, from an internal link to an overlay link, and from an overlay link to an internal link. When an edge device forwards between internal links **34**, **36** within its network site, it operates like a traditional L2 switch or L3 router (FIG. 2). The edge device **12** forwards a unicast, multicast, or broadcast packet from one of the internal links **34**, **36** to the overlay link **32** when IS-IS has put the logical port of the overlay interface in the MAC table **30** for the corresponding unicast or multicast MAC address. L2 traffic which traverses the overlay network **26** to reach its destination is prepended with an IP header that ensures the packet is delivered to the edge devices that provide connectivity to the L2 destination in the original MAC header. In one embodiment, a packet as received from the internal interface **34**, **36** is not changed other than to remove a preamble and FCS from the frame. The IP header, outer MAC header, and physical port the packet is to go out is preferably all cached in hardware. This is so all the information required to physically forward the packet is together to easily prepend and send data at high rates. The IP addresses and the outer MAC addresses are all preferably provided and stored for the hardware by the control-plane software.

When a packet **40** is received on the overlay interface **32**, it is decapsulated to reveal the inner MAC header for

forwarding. The inner MAC SA (source address) and DA (destination address) are used for the MAC table lookup described below.

FIG. 2 illustrates an example of MAC table **30** for edge device **12** having a core facing IP address of IPB and in communication via overlay interface **32** with a core network connected to another network site comprising an edge device having an IP address of IPA and a host H1 (not shown). Host H1 is reachable through edge device IPA via overlay network O1. The table **30** includes a number of destination/link pairs (or destination/next-hop pairs in the case of an L3 edge device) for forwarding between internal links (E1, E2), from internal links to overlay link **32**, and from the overlay link to the internal links. The MAC table **30** is populated using the address information received in LSPs and GM-LSPs at the overlay interface **32** and packets received at internal interfaces E1, E2, as previously described with respect to advertising addresses.

The first pair (H2/E1) in table **30** specifies that host H2 is reachable through link E1. Another host H3 at the network site is reachable through internal interface **36** (E2). Thus, another entry in the table is H3/E2. Since host H1 is reachable via the overlay network O1 through the edge device with IP address of IPA, the other entry in the table is H1/O1:IPA.

The table **30** provides information used by the edge device **12** in forwarding packets received at its interfaces. For example, based on a lookup at table **30**, packet **38** received at interface E1 and destined for host H1 is encapsulated in IP packet **40** with an IP header having a destination address of IPA. Packet **40** is transmitted from link **32** to the core network where it is routed (in the case of a routed core) to edge device IPA. A packet received at link **32** and destined for host H2, is stripped of its overlay IP header and forwarded to interface E1 in generally the same manner as a conventional bridge would forward a packet at L2.

It is to be understood that the table **30** shown in FIG. 2 and described above is only one example of an information base mapping addresses, and that other configurations may be used without departing from the scope of the invention. There may be cases where a VLAN will have some MACs that will be advertised and forwarded over the overlay network and others that will have their packets forwarded natively on physical interfaces. This can be controlled by policy configuration on the edge device. When a VLAN is enabled, an edge device begins advertising locally learned MAC addresses in IS-IS. If the MAC needs to be connected through the core natively, a network administrator can set up a router-filter based access-list to deny advertising the MAC. This may be used, for example, in shared services for an application.

Encapsulation

As previously discussed, OTV encapsulates traffic into an IP header which is natively handled in the IP or L2 core. Thus, rather than having logical interfaces at the edge devices representing tunnels going to different remote sites, the edge device encapsulates traffic based on its destination and transmits it from its overlay interface. Once the packet is received at the core network, core routing (in the case of a routed core) takes care of the traffic. The following describes actions that take place at the edge **15** device when an L2 frame is received that needs to be encapsulated in IP to be sent on the overlay network.

If the frame exceeds the MTU (Maximum Transmission Unit) of the physical outgoing link for the overlay network, the packet will be IP fragmented on an egress edge device and reassembled on an ingress edge device.

In one embodiment, the frame is IEEE 802.1Q encapsulated and three priority bits (IEEE 802.1p) in the Ethernet header are copied to a 3-bit IP-precedence field of the IP header. The remainder of the DSCP (Differentiated Services Code Point) field is padded with zeros, thus using Class Selector Code Points as defined in IETF RFC 2474 (“Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers”, K. Nichols et al., December 1998). Alternatively, a one-way mapping table from IEEE 802.1p to DSCP may be used instead of a direct copy.

The TTL (time to live) field in the IP header is preferably a configuration option, with a default value of 128, for example. The IP total length field is recomputed and the IP header checksum calculated. An IP option, such as ESP (Encapsulating Security Payload) or AH (Authentication Header) extension headers may be placed after the IP header so that IPsec (IP Security) can be used.

One example of encapsulation that may be used at the OTV edge device is described in IETF RFC 3378 (“EtherIP: Tunneling Ethernet Frames in IP Datagrams”, R. Housley et al., September 2002). The format includes an IP header, EtherIP header, and an encapsulated Ethernet frame. The EtherIP header includes the protocol version. Other types of encapsulation that may be used at the OTV edge device are EtherIP, GRE (Generic Routing Encapsulation) and L2RPv3 (Layer 2 Tunneling Protocol Version 3).

Edge Device Placement Examples

FIGS. 4A-4E illustrate examples of OTV edge device placement within network site 20 of FIG. 1. In each example, the network 20 includes an edge device X1 (12) and one or more routers (R1, R2) and switches (S1, S2, S3). The network 20 also includes hosts H1, H2, H3, connected to switch S1, switch S2, and router R1, respectively. The link leaving the network cloud 20 is the physical link to the core infrastructure.

In FIG. 4A, the edge device X1 is a border switch at the network edge. The network 20 includes hosts H1, H2, H3 and a router R1 on a switched network which includes switches S1, S2, and X1 and extends to the edge of the site. The edge device X1 is attached by the exit link to an L3 infrastructure and is inline with the data path. All host MAC addresses and router MAC addresses are advertised on the overlay network 26. When hosts in the switch network join multicast groups, the multicast MAC addresses are advertised on the overlay network 26. When host H3 joins a multicast group, a PIM (Protocol Independent Multicast) join-prune message is sent by router R1 to other routers on the extended multisite switch network.

FIG. 4B illustrates an edge device X1 one-hop from border switch S3, which is on path to edge. In this configuration, the edge device X1 is inline with the data path but the exit device is a L2 switch connected to an L2 metro Ethernet core. The advertisement for unicast and multicast MAC addresses is the same as described above with respect to FIG. 4A.

FIG. 4C illustrates an edge device X1 one-hop from border switch S3, which is not on path to edge. The configuration is similar to that described above for FIG. 4B except the edge device X1 is out of the data path.

FIG. 4D illustrates an edge device X1 one-hop from border router R2, which is on path to edge. This case is a typical configuration when two routers are connected together via a switched network. In this case there are also hosts H1, H2 on the switched network. The OTV edge device X1 is in the data path so it hears all packets that are going from hosts H1, H2, H3 to exit router R2, host to host within the site, and router R1 to exit router R2. In this

example, the edge device X1 can prefer intra-VPN traffic to take the overlay path versus the outside more public path.

FIG. 4E illustrates an edge device X1 that is a border device with router R2 one hop away. In this configuration, exit router R2 can be connected to either a L2 or L3 core infrastructure with the edge device X1 in the data path. The only MAC address that the site 20 will advertise is the MAC address of the exit router R2.

FIGS. 5A-5D illustrate various redundant network configurations. In each example, network site 20 includes one or more edge devices X1, X2 and routers R1, R2. FIG. 5A illustrates an edge device X1 with multiple external connections. In FIG. 5B a border router R1 has multiple external connections and edge device X1 has an L2 link to the border router. In FIG. 5C, the network includes multiple edge devices X1, X2, each with an external and overlay connection. The network shown in FIG. 5D includes multiple border routers R1, R2, each with external connections. Each edge device X1, X2 has a L2 link to each border router R1, R2, respectively, and a link to the overlay network 26.

It is to be understood that the network configurations shown in FIGS. 4A-4E and 5A-5D and described above are only examples and that an edge device configured for OTV may be used in other networks having different configurations and devices.

Overlay Transport Virtualization Process and Examples

FIG. 6A is a flowchart illustrating an example of the OTV process for packets received at the network device 12 from an internal link. At step 60, a first edge device 12 located within a first network site 20 receives MAC addresses for network devices (e.g., hosts) in one or more other network sites (e.g., second network site 22) within its VPN (FIGS. 1 and 6A). The edge device 12 maps each of the MAC addresses to an address of an edge device located in the network site of the other network device (step 61). The addresses are mapped in a table which also contains address information (learned either in the data-plane in the case of an L2 edge device, or in the cIGP in the case of an L3 edge device) for network devices in the same network site as the first edge device. At step 62, the first edge device 12 receives a Layer 2 packet from one of the network devices located at the same network site as the first edge device. The edge device 12 performs a lookup in table 30 for a link corresponding to the MAC destination address in the packet (step 63). If the destination is in the same network site as the edge device, the edge device forwards the packet on an internal link identified in the table (steps 64 and 65). The decision as to whether the destination is in the same network site as the edge device is implicit in the lookup. If the lookup yields a normal next-hop, the destination is in the same network. If the lookup yields an encapsulation, the packet is in a different network. If the destination is in another network site, the packet is encapsulated with the IP header containing a destination address (DA) of the edge device identified in the table (steps 64 and 66). The encapsulated packet is then forwarded to the core network (step 67), where it is forwarded to the appropriate edge device.

FIG. 6B illustrates a process wherein the packet received at the edge device 12 is from an overlay interface and destined for a network device within the same site as the edge device. At step 68, the edge device 12 receives a packet on overlay link 32 (FIGS. 2 and 6B). The packet is decapsulated at step 69. The MAC destination address is used to look up the internal link 34, 36 in table 30 (step 70). The packet is then forwarded onto the internal link identified in the table (step 71).

The following provides a number of examples describing unicast and multicast flow between L2 sites, L3 sites, and a combination of L2 and L3 sites. Broadcast packet flow, MAC mobility between network sites, and the partition and merging of network sites are also described. In the following examples the core network is an L3 network, however, it is to be understood that the core network may also be an L2 network, as previously discussed.

Unicast Packet Flow Between Layer 2 Sites

FIG. 7 illustrates unicast packet flow between two L2 sites. A Layer 2 site is one that is mostly made up of hosts and switches. Routers may exist but the majority of the topology to the edge devices are L2 switched. The number of MAC addresses advertised on the overlay network are all the hosts and routers connected to the L2 devices at the site.

The network shown in FIG. 7 illustrates any-to-any connectivity at the MAC level between any MAC devices. The network includes an L3 core network 72 in communication with two network sites 74, 76 having mostly L2 switches deployed. Network site 74 includes edge device X71, switches S71, S72, and router R71. Host 71 is connected to switch S71, host H72 is connected to switch S72 and host H73 is connected to router R71. Site 76 includes edge device X72, switches S73, S74, and router R72. Host H74 is connected to switch S73, H75 is connected to switch S74, and H76 is connected to router R72. The routers R71, R72 are connected to their respective hosts with an L3 link and the switches, routers, and edge devices are connected via L2 links. Since hosts H73 and H76 are supported by routers, their MAC addresses are not known to the switched network.

Overlay network 78 is shown connected to edge devices X71 and X72. The L3 core network 72 operates provider unicast/multicast routing protocol (pUMRP), the two customer networks 74, 76 run customer unicast/multicast routing protocols (cUMRP), and the overlay network 78 runs an overlay unicast routing protocol (oURP). As previously described, subnets and VLANs span across all sites. Each site has its own Spanning Tree and there is no L2 flooding or learning on the overlay network 78.

Before packet flow begins, edge devices X71 and X72 are configured with an IP multicast group address and IS-IS neighbor authentication key. The group address is allocated by the L3 core network 72 and is used to realize the control-plane component of the overlay network 78. The IS-IS neighbor authentication key allows for only intended edge devices to peer with each other. When IS-IS Hello packets are not accepted due to authentication failure, IS-IS adjacencies are not established and therefore no MAC routes are computed through the non-authenticated neighbor.

The following describes a packet flow sequence from host H71 in site 74 to host H74 in site 76 in the same VPN. Host H71 wants to send an IP packet to H74's IP address but does not know host H74's MAC address. H71 sends an ARP packet (broadcast MAC packet) which is forwarded by the switches (S71) in the source site 74 on the sites Spanning Tree. This causes all switches, including edge device X71 to learn about H71's MAC address. Edge device X71 encapsulates the broadcast packet into an IP multicast packet to be transmitted to all sites of the VPN. The edge device sends a single copy of the packet and allows the L3 core network 72 to replicate on its behalf. Edge device X71 advertises H71's MAC address in an IS-IS LSP since it has learned that H71 is local to its site. When edge device X72 receives the broadcast packet, it forwards the packet on the Spanning Tree for its site. This allows the packet to get to H74 so it can reply to the ARP request. When edge device X72

receives the IS-IS LSP from X71, it computes that the H71 MAC address is reachable through the IP address of X71. This is stored in the MAC table in edge device X72.

Host H74 replies to the ARP request by sending a unicast packet to H71's MAC address. The switches (S73, S74) in H74's site know where H71 is since they learned H71's address when they saw the broadcast ARP request packet on the Spanning Tree.

Switch S73 forwards H74's unicast packet destined for H71 to edge device X72. Edge device X72 receives the ARP reply packet and knows that H71 is reachable via the overlay network 78. Edge device X72 encapsulates the frame in an IP unicast packet addressed to edge device X71's IP address. The core network 72 delivers the packet to X71.

When edge device X71 receives the IP packet, it decapsulates the packet and finds that the H71 MAC address is the destination address in the frame header. Edge device X71 knows that host H71 is reachable via an internal link, so it forwards the frame out that link. Switch S71 receives the frame and forwards it directly to H71.

In the above example, if H73 sends packets to H76, R71 and R72's MAC address are used for forwarding through the L2 cloud across sites. Unicast Packet Flow between Layer 3 Sites A Layer 3 site is one that is mostly made up of routers connecting to hosts via switches. The majority of the topology to the edge devices are L3 routed. The number of MAC addresses advertised on the overlay network are limited to the router devices at the site. When a site runs as a L3 site, routers located in different OTV sites will look like they are adjacent to each other on the same subnet. Since an OTV edge device is a L2 device, all router protocol traffic and packets directed by the router to another router's MAC address for forwarding data packets, get switched by the edge device over the overlay network. In this case, there is one client IGP running across all sites within the same VPN.

FIG. 8 illustrates unicast packet flow between two sites 84 which are predominately made up of routers and connected through a L3 core network 82. There are very few MAC addresses carried in the switched network and transported on the overlay network. The topology allows the routers to look adjacent, as if they were on the same physical subnet. Site 84 includes edge device X81, routers R81, R82, and R85 and hosts H81, H82, and H83. Site 86 includes edge device X82, routers R83, R84, R86, switches S81, S82, and hosts H84, H85, and H86. All routers run a customer routing protocol (cIGP) to learn about all subnets inside the IGP domain. Routers R81, R82, R83, and R84 are configured on the same subnet and packet reception is realized by the inter-site switched network.

The following describes a packet flow sequence from host H81 in site 84 to host H84 in site 86. Host H81 first sends a packet to H84's IP address. H81 knows H84's address is off the subnet, so it sends the packet to its default router, which is router R81. The packet from host H81 is therefore MAC addressed to router R81. Router R81 knows via the cIGP that the next hop to reach the subnet of host H84 is via router R83. It sends the packet encapsulated with R83's MAC address as the destination address in the frame.

Edge device X81 receives the frame and based on previous packets sent from R83 (the routing protocol was sending and receiving packets to and from this MAC address), edge device X81 knows the next-hop for R83's MAC address is via X82's IP address through the overlay network 88. The core network 82 delivers the packet to X82 where it is decapsulated and sent on the X82-R83 link via MAC learning from a previous packet sent with R83's MAC address. Router R83 routes the packet to router R86 which

forwards to host H84 via normal routing which is unaffected by the edge device or overlay network.

Multicast Packet Flow Between Layer 2 Sites

In one embodiment, an ASM/Bidir multicast group is provided for each VPN the edge device supports. An (S, G) route in the core uses G as the data group used at the site, with S being the edge device. The edge device encapsulates a client multicast packet in a core-deliverable multicast packet. In one embodiment, joining either ASM/Bidir or SSM (Source Specific Multicast) groups is done using IGMPv3/MLDv2 in the edge devices.

FIG. 9 illustrates multicast packet flow between L2 sites in a VPN. The network includes an L3 core network 92 in communication with three L2 sites 94, 96, 97. An overlay network 98 connects edge devices X91, X92, X93, at sites 94, 96, 97, respectively. Site 94 also includes switches S91, S92, connected to hosts H91 and H92, respectively, and router R91 supporting host H93. Site 96 includes switches S93 and S94, connected to hosts H94 and H95, respectively, and router R92 connected to host H96. Site 97 includes switches S95 and S96 connected to hosts H97 and H99, respectively, and router R93 coupled to host H98. The routers are connected to their respective hosts through L3 links. The switches and routers are connected via L2 links. The edge devices are connected to the L3 core network via L3 links. The sites support IGMP/MLD-snooping, however, IGMP/MLD snooping and head-end replication are not performed over the overlay network 98. Packets go only to sites with group members.

The following describes a multicast source S (host H91) at network site 94 sending data to two different multicast groups G1 and G2. G1 members are only in site 97 and G2 members are in both sites 96 and 97.

Host H99 first sends an IGMP/MLD report for the two multicast groups it wants to join (G1 and G2) and host H95 sends an IGMP/MLD report for G2. Switch S96 (an IGMP/MLD snooping switch) forwards the IGMP/MLD report from H99 to router ports and querier ports. Edge device X93 hears the IGMP/MLD reports and does not forward them on the overlay network 98, since STP and IGMP/MLD-snooping packets do not occur in the overlay network. Edge device X93 advertises the G1 and G1-MAC in its GM-LSP so that edge devices X91 and X92 can learn what groups are active in the overlay network 98. If the edge device in a multicast source site knows the IP group address being joined by a remote site, it can avoid forwarding packets for another group which maps to the same MAC address.

Edge device X93 sends an IGMPv3/MLDv2 (X1, G1) and (X2, G2) report onto its core link so that multicast data is not encapsulated with the same group address as the IS-IS messages. The core network 92 builds SSM trees for group G1. If a source in X91 or X92's site starts sending, packets go only to X93's site. There is no coordination of group addressing required between the edge devices and the core because the (S, G) is unique as long as the source S is unique. The source S is unique due to the address allocation configured in the edge devices. In this case, S is the edge device's IP address and G is used in the cMRP as well as the pMRP.

Source H91 now sends a packet to group G1. Switch S91 sends a packet on router ports which X91 will be transit for. Edge device X91 learns host H91's MAC address on the X91-S91 link. X91 determines that the packet is being sent to group G1 and notices that it has a MAC entry, learned via IS-IS for group G1. The interface in the oif-list for the entry is the overlay network 98. Therefore, edge device X91 will encapsulate the packet in an IP multicast packet where the

source address is X91 and the destination address is the same as the address the source sent the packet to (group G1).

Edge device X91 then sends the packet on the link into the core network 92. The core network 92 forwards the multicast packet based on (X91, G1) which gets the packet to site 97 with the single receiver. When source S sends to group G2, the same procedure is used except edge device X91 encapsulates the packet as (X91, G2).

The core network 92 will then deliver the packet on the SSM-based multicast tree for (X91, G2), which delivers the packet to both sites 96, 97 (where receivers exist in both sites). The amount of state that is created in the core network 92 is based on the number of groups the VPN joins times the number of authoritative edge devices in all sites of a VPN.

An optimization may be implemented to create SSM trees only to the sites where there are active sources. To achieve this, edge devices advertise in IS-IS (S, G) pairs which identify an active source S sending to group G. Then, when a receiver site sends IGMP/MLD joins for a group G, it does so only for the active sites sourcing to group G.

In one embodiment, rather than advertising the source address, the groups which are being sourced can be identified. This allows receiver sites to build SSM trees to only the source sites.

Also, the source site can be determined even when (S, G) pairs for active sources are not advertised, but when a receiver site IGMP/MLD joins an (S, G) pair. However, the source's IP route would have to be known by the edge device. The edge device does not have this information if it stores only unicast MAC addresses, multicast MAC addresses, and IP group addresses. When an edge device is co-located with an IP router, this information is known locally from the IP routing table maintained in the edge device for the cIGP. This optimization may also be used to reduce multicast state in the core network.

Multicast Packet Flow Between Layer 3 Sites

FIG. 10 illustrates multicast packet flow between L3 sites. As described below, the L3 sites run PIM across the L2 overlay network. An L3 core network 102 is in communication with three L3 sites 104, 106, 107. Site 104 includes edge device X11, routers R11, R12, R15, and hosts H12 and H13. Site 106 includes edge device X12, routers R13, R14, R16, switches S11, S12, and hosts H14, H15, and H16. Site 107 includes edge device X13, switches S13, S14, router R17, and hosts H17, H18, and H19. Routers R11, R12, R13, R14, and R17 and hosts H17 and H19 look adjacent to each other on the same common subnet. These five routers IGP and PIM peer with each other. If hosts H17 and H19 join groups, they are treated like directly connected receivers and if they send multicast packets, they are treated as directly connected sources. The routers IGP and PIM peer with each other. If the hosts join groups they are treated like directly connected receivers and if the hosts send multicast packets they are treated as directly connected sources.

The following example starts with host H18 joining groups G1 and G2 in site 107 and host H14 joining group G2 in site 106. The Rendezvous Point (RP) is placed at router R12 and the source is at host H11 at site 104. Host H18 sends IGMP/MLD reports for groups G1 and G2. Router R17 is the PIM designated router (DR) on the host H18 subnet. Router R17 therefore sends a (*, G1) and (*, G2) PIM join-prune message to the next-hop router toward the RP. Unicast routing will identify this as R12 (which happens to be the RP in this example, but does not need to be).

Edge device X13 then performs PIM-snooping at the site, but not on the overlay network 108. X13 takes the (*, G1) and (*, G2) routes from the PIM join-prune packet and sends

an IGMPv3/MLDv2 (X11, G1) and (X11, G2) into the core. In this case, different than the IGMP/MLD-snooping case, the edge device knows where the PIM join-prune message is going, so it does not have to IGMPv3/MLDv2 join to all edge devices. Edge device X13 translates the upstream router address in the PIM join-prune message into a MAC address, and then does a MAC address lookup to see who has advertised it. In this example, router R12's MAC address is being advertised by edge device X11, so edge device X13 can IGMP/MLD join (X11, G1) and (X11, G2).

When a source from X12's site 106 sends a packet, edge device X13 can later join those SSM entries. Edge device X13 also advertises G1, G1-MAC, G2, and G2-MAC in its GM-LSP. When router R12 receives the PIM join-prune message, it places the link to router R17 (R12-X11 link) in its L3 oif-list. This is the first oif that is on the shared-tree for both entries (*, G1) and (*, G2).

Source H11 now sends a multicast packet to group G1. Router R11 is the PIM 10 DR for the subnet so it will PIM register encapsulate the packet to router R12. R12 will decapsulate the packet and forward it down the shared tree, in this example, out the X11-R12 link. Edge device X11 knows that G1-MAC is in the oif-list for the MAC table entry since edge device X13 advertised it in its GM-LSP, so it will encapsulate the packet with addresses (X11, G1) which the core is ready with a SSM-based tree already built.

For the G2 case, edge device X11 encapsulates in (X11, G2) and the core delivers the packet to both sites 106, 107. The shared-tree has now been built and the first packets have traveled on the shared-tree. The following describes how the source tree is setup. Once router R17 receives a packet on the (*, G1) shared-tree from source host H11, router R17 joins the source-tree.

In this example, router R17 sends an (H11, G1) PIM join-prune message. The message is sent to router R11 since it is on the shortest path to the source H11. Edge device X13 then again PIM snoops on the PIM join-prune message and interrogates the upstream router address and realizes the next-hop router R11's MAC is via X11's IP address. Since it has already IGMP/MLD joined an (X11, G1) for the site's (*, G1) entry, it won't have to for the site's (H11, G1) entry. So in this case, both entries (H11, G1) and (*, G1) can share the same SSM-based state in the core.

Router R17 prunes (H1, G1) off the shared-tree but the edge device X13 knows there are source trees using (X11, G1) so it will not IGMP/MLD leave (X11, G1) until all source trees are left at the PIM level.

Multicast Packet Flow in a Hybrid VPN

FIG. 11 illustrates a hybrid VPN comprising L2 site 117 and L3 site 114 connected by core network 112. Site 114 includes edge device X21, routers R21, R22, and R25, and hosts H22, H23. Site 117 includes edge device X23, switches S23, S24, and hosts H28, H29.

Translation is performed to and from IGMP-snooping and IS-IS GM-LSP advertisement. For example, if host H28 in L2 site 117 joins group G1, IGMP-snooping is performed along all switches. However, the IGMP report is not sent on the overlay network 118 by edge device X23. Edge device X21 learns about group G1 from IS-IS, but routers R21 and R22 need to know the group address so they can join the shared-tree in the source domain. Edge device X21 therefore needs to originate IGMP reports that it learns from its GM-LSP. Since the GM-LSP contains both an IP group address and a MAC group address, the edge device has all the necessary information to generate an IGMP report.

If host H28 is a source in the L2 site 117 and host H22 a group member in the L3 site 114, the IGMP-snooping logic

in the L2 site would forward all data to router ports so that the edge device X23 will see such packets. Edge device X23 knows which groups to forward packets based on the GM-LSP advertisement originated by edge device X21.

5 Broadcast Packet Flow

A broadcast packet originated at an OTV site needs to be delivered to all sites of the same VPN. In one embodiment, this is accomplished with ASM/Bidir group encapsulation. This is the same group used by IS-IS for the oUMRP.

10 A broadcast packet sourced in a site, gets to all edge devices because each edge device is on the spanning tree. In order to avoid duplicates in the overlay network when there are multiple edge devices, the authoritative edge device for the VLAN is the only edge device that forwards the packet on the overlay network. The authoritative edge device prevents duplicate packets from going into the site because it is the only edge device forwarding the packet into the site. Once sent into the site, the packet gets to all switches on the site specific Spanning Tree.

20 Since only the authoritative edge device can forward the broadcast packet in or out of the site, broadcast loops are avoided as any non-authoritative edge device receiving a broadcast packet on its internal interface will not forward the broadcast packet onto the overlay network.

25 OTV may reduce the amount of broadcast traffic on the overlay network. For example, the OTV device 12 can snoop on ARP requests and cache the IP address to MAC mapping. If other hosts ARP for the same foreign IP address, the OTV device can ARP reply locally rather than send the broadcast traffic over the overlay network.

30 Alternatively, the OTV device 12 can put IP_address-to-MAC_address pairs in the oUMRP. The edge device 12 can then respond when an ARP request arrives for an IP address in the link state database. In this case, no ARP packets are sent over the overlay network.

MAC Mobility

In order to support server virtualization in Data Center networks, applications migrate from one physical server to another very quickly. Along with the application migration, the MAC address associated with the application also migrates. Since there is a single authoritative edge device for all traffic within a VLAN and IS-IS is explicitly advertising remote MAC addresses, the edge device knows when the MAC is remote and when it is no longer remote. If an authoritative edge device has a MAC address stored in the MAC forwarding table which points to the overlay interface, it means that an edge device in another site has explicitly advertised the MAC as being local to its site.

Therefore, any packets coming from the MAC address will be coming from the overlay network. Once the MAC address is heard on an internal interface, it has moved into the site. Since it has moved into a new site, the authoritative edge device in the new site is responsible for advertising it in its LSP.

55 It is the authoritative edge device in the site the MAC address has moved to that knows immediately that it has moved from arrival of a data packet from the MAC address on an internal interface. The authoritative edge device in the site the MAC has moved from does not know the move happened and still has the MAC address in its MAC table pointing to an internal interface. The authoritative edge device in the site from which the MAC address has moved from, may find out that the MAC is no longer in its site from in a number of different ways. For example, the authoritative edge device can hear packets from the MAC address coming from the overlay network or it can be told via IS-IS. The second method is preferred because if there is no traffic

coming from the moved MAC address to the site it moved from, no corrective action would be taken, and other remote sites will still think the MAC is in the old site.

FIG. 12 illustrates a MAC mobility example in which H1 MAC moves from switch S31 in network site 122 to switch S32 in network site 124. The sites are connected by L3 core network 120 and overlay network 126. Site 122 includes edge devices X31 and X32 and site 124 includes edge devices X33 and X34. When host H1 20 and H2 have been chatting, the state in the MAC tables appear as shown by the hosts listed at each interface in FIG. 12. Changes following the H1 MAC move are shown in italics and indicated with dashed arrow lines. In a redundant edge device configuration, local MACs are advertised with best-cost by one edge device so there is one way in and out of a site. Support is active-backup for MACs within a VLAN but active-active across VLANs. When H1 moves, edge device X33 detects the move since it hears packets on its X33-S32 link. Edge device X33 changes the MAC table for the H1 MAC and advertises H1 in IS-IS. Edge device X31 sees H1 MAC advertisement from X33 and withdraws advertising H1 MAC from its LSP. Site 122 as well as other sites now know H1 is reachable via edge device X33.

Site Partition/Merging

When a site has two or more edge devices connecting the site to the overlay network, a site can be partitioned but still connected to the overlay network and can be reachable as two independent entities to remote sites in the VPN. When this happens, each edge device loses its internal site peering adjacency with the other. Each site elects a new site-id and the site-id is advertised in IS-IS Hello messages on the overlay network. At the same time, if this site had exactly two edge devices, each would become authoritative for the entire VLAN-ID range. This means that there will be new MACs that need advertising from both partitioned edge devices. When the site partitions, the MAC table is preferably cleared and an STP topology change notification sent, if STP is running at the site, so that unicast and multicast MAC addresses can be relearned and advertised into IS-IS. Also, since the site partitioned, there will be two independent Spanning Trees built, one for each site.

Sites may merge, for example, when a link is inadvertently connected to two switches or routers that are intentionally configured to be in different sites. Also, sites may merge during a rehealing after a partition, as previously described. In the case where the two sites are L2 sites, two individual Spanning Trees become one with one new root bridge is elected. All OTV edge devices will notice each other and there will be new authoritative edge device election for each VLAN-ID range. When an edge device was authoritative and becomes non-authoritative, it needs to remove all MAC entries that point out the overlay network from its MAC table and withdraw all unicast and multicast MAC advertisements from its LSPs.

FIG. 13 illustrates merging of two L2 sites 132 and 134 into network site 136. Site 132 includes edge device X41, switches S41 and S42, router R41, and hosts H41, H42, and H43. Site 134 includes edge device X42, switches S43 and S44, router R42, and hosts H44, H45, and H46. When the sites merge at L2 by connecting link 135, two single-home sites become one multi-homed site and two separate Spanning Trees become one. In this example, edge devices X41 and X42 IS-IS peer with each other over both the overlay network 138 and the internal switched network. On the internal peering, they elect a site-id, which is the system-id of the highest priority edge device. Both advertise the site-id

on the overlay peering link so they, as well as other edge devices in the VPN, detect they are at the same site.

Load Balancing and ECMP

The following describes an example for load balancing and ECMP (Equal Cost Multipath) across a VLAN. This can be used to provide optimal multi-homing for OTV sites. FIG. 14 illustrates an example of a network comprising two sites 142, 144 connected by core network 140 and overlay network 146. Site 142 includes edge devices X51 and X52 and switch S51 (H1). Site 144 includes edge devices X53 and X54 and switch S52 (H2). In this example, edge device X51 is an authoritative edge (AE) device for forwarding traffic from H1 to H2 and receiving traffic from H2 destined for H1. Edge device X53 is an authoritative edge device for receiving traffic from H1 destined for H2 and forwarding traffic from H2 to H1. Determination of the edge device is described below.

The authoritative edge device handles all traffic for a given remote MAC address. A non-authoritative edge device is any edge device that is not the authoritative edge device for a given MAC address. The authoritative edge device is determined using a hash algorithm based on remote MAC address and local ordinals. Each edge device has an ordinal assigned thereto. Unicast forwarding involves the choice of a local egress device and a remote ingress device. The local egress device is determined based on a destination address (DA) hash. The remote ingress device is determined locally based on a source address (SA) hash.

Referring again to FIG. 14, the following describes symmetric hash decisions for a unicast hash example. Calculations are first performed at site 142 to determine the local (site 142) egress authoritative edge device and the remote (site 144) ingress authoritative edge device for use in forwarding data from H1 (source address (SA)) to H2 (destination address (DA)). In order to determine the local egress device, a local hash function (f) is performed on the destination address (MAC address for H2 at the remote site) and local ordinal (LO) for site 142 (O1), as follows: AE(H2) H1→H2=f(DA,LO)=f(H2,O1)=X51

The remote ingress device is identified by performing a remote hash function (f) on the source address (MAC address for H1 at the local site) and remote ordinal (RO) for site 144 (O2), as follows: AE(H1) H1→H2=f(SA,RO)=f(H1,O2)=X53

The following calculations are performed at site 144 to determine the local egress device and remote ingress device for use in forwarding data from H2 (source address (SA)) to H1 (destination address (DA)). A local hash function (f) is performed on the destination address (MAC address for H1 at the remote site) and local ordinal for site 144 (O2), as follows: AE(H1) H2→H1=f(DA,LO)=f(H1,O2)=X53

The remote ingress device is identified by performing a remote hash on the source address using a remote ordinal for site 144, as follows: AE(H2) H2→H1=f(SA,RO)=f(H2,O1)=X51 The remote MAC address (DA) may be learned by data or control plane learning. The local ordinals are elected and known locally. The local MAC address (SA) is learned by conventional bridging. The remote ordinals are remotely defined parameters which are obtained using control plane learning. All edge devices in a site advertise local MAC addresses to the overlay to enable ECMP forwarding in the core network 140. Nonauthoritative edge devices wait for the authoritative edge device to advertise a specific MAC address before they advertise reachability for the MAC address.

FIG. 15 illustrates an example of broadcast hashing for multi-homed sites. Network sites 152 and 154 are connected

by core network **150** and overlay network **156**. Site **152** includes edge devices **X61** and **X62** and switch **S61** (H1). Site **154** includes edge devices **X63** and **X64** and switch **S62** (H2). In this example, edge device **X61** is an authoritative edge device and edge device **X62** is a broadcast forwarder (BF) at site **152**. Edge device **X63** is both an authoritative edge device and a broadcast forwarder at site **154**. The broadcast forwarder handles all broadcast traffic for a given source MAC address. Broadcast forwarding involves the choice of a local egress device and a remote ingress device. The broadcast forwarder is determined based on source MAC address for both origin and destination sites, as described below.

The local egress BF (origin site **152** to core **150**) is determined at the origin site by performing a hash of source MAC address (local) and local ordinal. The ingress BF (core **150** to destination site **154**) is determined at the destination site by performing a hash of source MAC address (remote) and local ordinal. At the destination site, the broadcast forwarder is the same as the authoritative edge device for the source address (AE(SA)) to avoid learning flaps of the source address at the destination site. B-cast traffic received on non-BF edge devices is not forwarded between core and site interfaces.

Referring again to FIG. **15**, the following describes symmetric hash decisions for a broadcast hash example. The local egress BF for forwarding broadcast from H1 (source address (SA)) at site **152** (local) to H2 at site **154** (remote) is determined by performing a hash function (f) on the source address and local ordinal for site **152** (O1), as follows: $BF(H1)_{origin} = f(SA, LO) = f(H1, O1) = X62$

The remote ingress BF is calculated using a source address equal to H1 and the local ordinal of site **154** (O2), as follows: $BF(H1)_{destination} = f(SA, LO) = f(H1, O2) = X63 = AE(H1)$. As noted above, the broadcast forwarder at the destination site is equal to the authoritative edge device for the source address.

In Layer 2 networks, multicast data traffic may always be forwarded towards local receivers and router ports. An OTV edge device, by periodically originating multicast router discovery packets, may masquerade as a router port. This will result in the OTV edge device attracting Layer 2 multicast traffic to itself. A “drop and punt to CPU” route may initially be installed to cover the entire multicast range. Once the control plane learns (VLAN, S, G) information from the punted packet, a more specific (Vlan, S, G) route may be installed in the forwarding engine.

When a VLAN is configured to be extended to provide OTV service over a service provider network, a (VLAN, *, 224.0.0.0/4) route may be installed into the forwarding engine. This installed route will result in the dropping of any multicast data packet hitting this entry. The dropped data packet may subsequently be copied to the device CPU.

Next, any new multicast data frames arriving on the Layer 2 port on the OTV edge device that hits the (VLAN, *, 224.0.0.0/4) forwarding engine entry may be copied to the device CPU. The copy may be received by the CPU and processed by a control plane component. The associated (VLAN, S, G) information of the multicast frame may be stored locally and then sent to the forwarding engine to either 1) drop the frame if no interested remote receivers exist or 2) encapsulate the frame with a provider data delivery group and sent to the service provider network.

The (VLAN, S,G) information, along with the selected data delivery group may be distributed using OTV Overlay Routing Protocol (e.g., OTV-ISIS) to remote sites. A subsequent data packet with the same (VLAN, S, G) informa-

tion may hit the entry (as the longest match) in the forwarding engine and either be dropped or encapsulated and sent to the service provider network.

The liveness of the (VLAN, S, G) streams may be tracked by the control plane by periodically learning the incoming data traffic rate. If the data source stops sending the (VLAN, S, G) stream or if the source moves to a remote site, it may be subsequently discovered by the CPU as incoming traffic will cease to increment. At that point, the CPU may withdraw the programmed (VLAN, S, G) entry from the forwarding engine.

In embodiments described in this specification, IGMP snooping may always be enabled on VLAN while OTV is enabled. IGMP snooping module may warn the user if not enabled. OMF may also always be enabled by IGMP snooping on VLAN if OTV is enabled. IGMP snooping module may warn the user if OMF is not enabled.

The punting step as described above may be rate-limited. In some embodiments, an OTV Routing Information Base (“ORIB”) may only add the following entries to a multicast FIB [forwarding information base] distribution memory (“MFDM”) where: 1) (V, *, 224.0.0.0/4) with a punt bit set and a null oiflist and 2) (V, *, 224.0.0.0/24) with only the overlay in the oiflist. Furthermore, all active source entries that are locally learnt may be added to the MFDM. Remote active source entries may not be added by ORIB to MFDM. Only the overlay interface may be in the oiflist. The overlay interface may be conditionally added by ORIB if it comes to know that there is interest for such traffic in the remote sites.

The multiroute adding of active source entries may comprise inheriting the oiflist of (V, *, *); (V, *, G), and (V, S, G) routes. In some embodiments, IS-IS may always do the adding with a multi-route flag set: 1) remote active source entries and 2) remote IGMP routes (V, S, G); (V,*, G), and (V, *, *). IGMP may always add local IGMP routes (V, S, G); (V,*, G), and (V, *, *) with a multi-route flag set. Finally, OTV may always add local active source entries with a multi-route flag set provided by the mcastfwd process).

If a component adds an active source entry with multi-route add flag to ORIB, ORIB may inherit oifs from the following set of routes (V, S, G); (V,*, G), and (V, *, *). When a component adds any of (V, S, G); (V,*, G), or (V, *, *) routes with multi-route add flag to ORIB, ORIB may inherit oiflist of the route to all matching active source entries.

FIG. **16** illustrates a flowchart describing embodiments of the present disclosure. In this described example, there may be no router ports in any site and a source SB1 may send data traffic for a group G1 from siteB. At step **1605**, IGMP snooping on both edge boxes may initially install (V, *, *) route information with a null oiflist into MRIB and ORIB. The method may proceed to step **x10** where an MRIB message may add the (V, *, *) entry to MFDM which goes into a server monitoring device. One such device for this example is referred to as Eureka.

The method may then proceed to step **1615** where ORIB adds the (V, *, *) entry with a punt flag to MFDM which goes into Lamira. At step **1620**, ORIB may also notify IS-IS about the (V, *, *) entry which may subsequently carry it to remote sites in GM-LSP and adds the entry to ORIB still with a null oiflist. At step **1625**, source SB1 in siteB may start to send traffic for group G1. This traffic will hit the (V, *, *) entry installed in Lamira on edge device B (“EB”) and get punted to an L2Netstack process.

Next, at step **1630**, L2Netstack may handle this information and inform OTV about (V, SB1, G1). At step **1635**, OTV may append delivery information (e.g., (DSB1, DG1)) to

this route and adds this entry to ORIB with a null oiflist. Then, at step 1640 ORIB may add this information to MFDM which goes into Lamira.

At step 1645 any further data packets (SB1, G1) may hit (V, SB1, G1) and get dropped as no oifs are present. Also, at step 1650, ORIB may notify IS-IS about the discovered source SB1 in an active source type length value (“TLV”). At step 1655, IS-IS may carry this information in GM LSP. After ISIS on remote site (in this case, siteA) gets it, it will update DRIB about it. Note that in some embodiments, the local source entry information may be informed by L2Netstack and the remote source entry information is informed by ISIS.

Next, at step 1660, ORIB may perform a lookup to find out if there is local interest for group G1. It looks up matching IGMP snooping installed routes in the following order: (V, S, G), (V, *, G), (V, *, *). The method may then proceed to step 1665 where it may be determined that none of the matching entries have IGMP oifs. Then, at step 1670 ORIB may assert that there is no interest in group G1 and hold the information (V, SB1, G1) [DSB1, DG1] and not process it further. It should be seen that all the traffic that is received by EB is not forwarded onto the overlay. In some embodiments, from 1670, the method may proceed to step 1705 or 1805 as described in further detail below in regards to FIGS. 17 and 18 respectively.

FIG. 17 is a flow chart illustrating steps continuing from step 1670 when source SB1 sends data traffic for group G1 from site B and subsequently, a router port may be discovered in site A. Step 1705 may comprise discovering a new router port by an edge device A (“EA”) in siteA on the interface ral adaptation layer. The method may then proceed to step 1710 where IGMP snooping may be employed on EA updates (V, *, *) with ral in the oiflist to MRIB and ORIB with a multi-route flag set.

Next, at step 1715, an MRIB message may update (V, *, *) with the new oiflist. This update may also go into Eureka. At step 1720, ORIB may update this entry to MFDM with the new oiflist that may also go into Lamira. At step 1725, ORIB may then walk its internal database of discovered source entries to see if it needs to process any such entries.

When ORIB encounters (V, SB1, G1) [DSB1, DG1], the method may proceed to step 1730. At step 1730 the ORIB may inherit ral from (V, *, *). The method may then move to step 1735 where ORIB may notify OTV and IS-IS about the ral as the oiflist for the discovered active source entry is not NULL. The method may then move to step 1740 where the OTV may inform IGMP of the added information. IGMP may send join messages into the core for (DSB1, DG1).

As there is local interest, the method may proceed to step 1740. At step 1740, OTV may add (DSB1, DG1) into a Multicast Routing Information Bank (“MRIB”). The addition may include a decap flag and overlay interface added to the oiflist. MRIB may subsequently add this information to MFDM.

At step 1745, ISIS may carry (V, *, *) entry with a router port bit set. At step 1750, ISIS on the remote site (site B) may get this and now update ORIB with overlay in the oiflist as the router port is identified with site A. ORIB may subsequently update this entry in Lamira. Next, the method may proceed to step 1755. At step 1755, ORIB may walk its internal database of discovered source entries to see if it needs to process any of the source entries.

When the method reaches step 1760, ORIB may encounter (V, SB1, G1) [DSB1, DG1]. At step 1760, the ORIB will inherit the overlay interface from (V, *, *). ORIB may then add this entry to MFDM as it is the local active source entry.

The method may then proceed to step 1765, where subsequent traffic (SB1, G1) may hit (V, SB1, G1). This subsequent traffic (SB1, G1) will be sent on the overlay using proper encapsulation. If the multicast data tree is setup, the encapsulated packets will reach to the EA.

As the encapsulated packets reach the EA, the method may proceed to step 1770. At step 1770, the encapsulated packets at the EA hit (DSB1, DG1) which has the decap flag. The method may then proceed to step 1775, where the packets may be decapsulated. This decapsulation may reveal (SB1, G1) information for the vlan V.

The method may then proceed to step 1780. At step 1780, additional lookup is done for (V, SB1, G1). The lookup may hit (V, *, *), which is installed in Eureka and subsequently sent to the oiflist. It should be understood that steps in the above described method may be similar even if the router port is discovered first in site A and then source SB1 starts sending data traffic for group G1 from site B.

FIG. 18 is a flow chart illustrating steps continuing from step 1670 when the Source SB1 sends data traffic for Group G1 from site B and subsequently, a host expresses interest in group G1 in site A. Step 1805 may comprise a host sending a (*, G1) report. This report would be attracted by EA. In some embodiments, OTV on an authoritative edge box may periodically generate PIM hellos on a site-facing interface. The method may then proceed to step 1810 where IGMP snooping may be employed on EA updates (V, *, G1) with the hardware adaptation layer (“hal”) in the oiflist.

Next, at step 1815, a MRIB message may add the snooped entry to MFDM with the new oiflist. This update may also go into Eureka. At step 1820, ORIB may hold this entry in an internal database. In some embodiments, ORIB should not add this route to MFDM. At step 1825, ORIB may walk its internal database of discovered source entries to inherit the hal to all matching source entries.

When ORIB encounters (V, SB1, G1) [DSB1, DG1], the method may proceed to step 1830. At step 1830 the ORIB may inherit hal from (V, *, G1). The method may then move to step 1835 where ORIB may notify OTV and IS-IS about the hal. The method may then move to step 1840. At step 1840, OTV may add (DSB1, DG1) into MRIB. The addition may include a decap flag and overlay interface added to the oiflist. MRIB may subsequently add this information to MFDM. The OTV may inform IGMP of the added information. IGMP may send join messages into the core for (DSB1, DG1).

At step 1845, ISIS may carry (V, *, G1) entry in its GM-LSP. At step 1750, ISIS on the remote site (siteB) may get this entry and now add (V, *, G1) to ORIB with overlay in the oiflist. ORIB does not add this information to MFDM. Next, the method may proceed to step 1855. At step 1855, ORIB may walk its internal database of discovered source entries to inherit the hal to all matching active source entries.

When the method reaches step 1860, ORIB may encounter (V, SB1, G1) [DSB1, DG1]. At step 1860, the ORIB will inherit the overlay interface from (V, *, G1). The method may then proceed to step 1865, where all subsequent traffic (SB1, G1) that hits (V, SB1, G1) will be sent on the overlay using proper encapsulation. If the multicast data tree is setup, the encapsulated packets will reach to the EA.

As the encapsulated packets reach the EA, the method may proceed to step 1870. At step 1870, the encapsulated packets at the EA hit (DSB1, DG1) which has the decap flag. The method may then proceed to step 1875, where the packets may be decapsulated. This decapsulation may reveal

(SB1, G1) information for the vlan V. The method may then proceed to step 1880. At step 1880, additional lookup is done for (V, SB1, G1).

In embodiments of L2 multicast source learning, IGMP may install (BD, *, *) when snooping has been enabled on a VLAN (BD). Activating Optimal Multicast Flooding (OMF) in IGMP may allow for this snooping to be active. In some embodiments, OMF will always be on in OTV edge boxes.

When IGMP installs (BD, *, *) in ORIB, other clients (e.g., IS-IS) will be notified. When IS-IS receives a GM-LSP with the router port bits, IS-IS may also add (BD, *, *) to ORIB with overlay interface in the oiflist. In some embodiments, for example at an L3 FIB (BD, *, *) may be installed by MFIB.

ORIB may set a 'copy to sup' flag for the (BD, *, *). In some embodiments, the flag may be set upon IGMP update, i.e., when snooping is enabled on VLAN. This approach may result in the Active Source TLV being announced right away. In some embodiments, the flag may be set on the IS-IS update, i.e., when IS-IS adds the overlay interface to the oiflist. This may occur after receiving a GM-LSP with a router port bits set. When the flag is set upon IS-IS update, the activesource may not be announced until after the discovery of remote receivers, which can potentially cause join delay.

IGMP may add (BD, *, *) for the local hosts. This route may have a 'local' bit set in ORIB. As it would simply be a local route, ORIB will not install it in MFDM. Similarly, IS-IS may add (BD, *, G) when a remote receiver is discovered and the remote receiver is interested in group G. IS-IS will add the overlay interface to the oiflist. ORIB may then install this route in MFDM with the 'copy to sup' flag.

When an L2 multicast packet hits the (BD, *, G), a copy may be sent to a Supervisor. Netstack-l2mcast, may install a (BD, S, G) in ORIB. ORIB may then install this route in MFDM without the 'copy to sup' flag. MFDM may send the to MFIB which in turn installs the route into L3 FIB. Note that for L3 netstack-l3mcast, it will talk to MRIB to install the route in MFDM. The above ORIB interaction is achieved in a similar manner.

If a source stops talking, (for more than a pre-determined period of time, such as sixty seconds) the associated routes will need to expire. In L3 this may be accomplished by MRIB polling MFDM to obtain network statistics. In some embodiments, polling may be employed in ORIB to ensure that routes may be expired.

If IS-IS removes (BD, *, G), all associated (BD, S, G) entries will need to be removed, unless the (BD, S, G) route was explicitly added by IS-IS. If the (BD, S, G) route was explicitly added by IS-IS, then IS-IS may explicitly remove it by referring to the 'local' and 'remote' routes stored in ORIB.

FIG. 1900 is a flow chart illustrating additional embodiments described herein. The method may start at step 1910. At step 1910, an L2 multicast data packet may hit the (BD, *, G). A copy of the data packet may be sent to the supervisor device. Netstack-l2mcast may install a (BD, S, G) route entry into ORIB. ORIB may then install this route in MFDM without the 'copy to sup' flag. MFDM sends the route to MFIB. MFIB may then insert the route into an L3 FIB.

The method may proceed to step 1920, where ORIB may notify its client devices of the (BD, S, G) route with a special flag for IS-IS. IS-IS may advertise this information in an Active Source TLV. The special flag may be used to distinguish from a (BD, S, G) route that IS-IS advertises in Group

Address TLV. These differently stored may be handled differently as described below.

The method may then proceed to step 1930. At step 1930, IS-IS may notice the special flag implemented in step d20. IS-IS will subsequently retrieve Delivery Group (DG) and Delivery Source (DS) for the (BD, S, G) route from OTV. IS-IS may then advertise the (DG, DS, BD, S, G) information in a new (or existing) active source TLV through a GM-LSP.

The method then proceeds to step 1940 where the remote IS-IS may receive the active source TLV. The remote IS-IS may then send the DG, DS, and associated (S, G) information to OTV. It should be understood that in this example, IS-IS will not install the active source information to the ORIB.

Finally, at step 1950, OTV may begin maintaining the DG, DS database. The OTV will discover the sources for the associated groups. If there are any local receivers for the groups, the OTV and will trigger IGMP joins to the associated (DS, DG) trees in the core. The determination of interested local receiver may be made by referring to the ORIB database using an ORIB API.

Embodiments of the present disclosure, for example, are described above with reference to block diagrams and/or operational illustrations of methods, systems, and computer program products according to embodiments of this disclosure. The functions/acts noted in the blocks may occur out of the order as shown in any flowchart. For example, two blocks shown in succession may in fact be executed substantially concurrently or the blocks may sometimes be executed in the reverse order, depending upon the functionality/acts involved.

While certain embodiments of the disclosure have been described, other embodiments may exist. Furthermore, although embodiments of the present disclosure have been described as being associated with data stored in memory and other storage mediums, data can also be stored on or read from other types of computer-readable media, such as secondary storage devices, like hard disks, floppy disks, or a CD-ROM, a carrier wave from the Internet, or other forms of RAM or ROM. Further, the disclosed methods' stages may be modified in any manner, including by reordering stages and/or inserting or deleting stages, without departing from the disclosure.

All rights including copyrights in the code included herein are vested in and are the property of the Applicant. The Applicant retains and reserves all rights in the code included herein, and grants permission to reproduce the material only in connection with reproduction of the granted patent and for no other purpose.

While the specification includes examples, the disclosure's scope is indicated by the following claims. Furthermore, while the specification has been described in language specific to structural features and/or methodological acts, the claims are not limited to the features or acts described above. Rather, the specific features and acts described above are disclosed as examples for embodiments of the disclosure.

At least the following is claimed:

1. A method comprising:
 - installing a virtual local area network (VLAN) route entry into a first routing information database associated with a forwarding engine of a service provider network comprising a first overlay transport virtualization (OTV) site, wherein installing the VLAN route entry comprises installing the VLAN route entry configured to cause the forwarding engine to:

27

drop a first multicast data packet hitting the VLAN route entry, and
 punt the first multicast data packet hitting the VLAN route entry to a processor associated with the service provider network;
 5 receiving, when the first multicast data packet hits the VLAN route entry, the punted first multicast data packet at the processor;
 learning, from the punted first multicast data packet associated route information comprising a first VLAN address, a first source address, and a first group address; and
 10 updating the VLAN route entry to include the associated route information, wherein updating the VLAN route entry comprises appending delivery information comprising a destination source address at the first OTV site and a destination group address at the first OTV site to the updated VLAN route entry, wherein updating the VLAN route entry further comprises updating the VLAN route entry being configured to cause the forwarding engine to one of the following:
 15 drop, when no remote receivers exist, any multicast data packet hitting the updated VLAN route entry, and
 encapsulate, when one or more remote receivers exist, any multicast data packet hitting the updated VLAN route entry with a provider data delivery group information and send the encapsulated multicast data packet to the service provider network.
 20
 2. The method of claim 1, further comprising:
 informing the first OTV site the updated VLAN route entry; and
 storing the updated VLAN route entry to the first routing information database.
 25
 3. The method of claim 1, further comprising:
 notifying an Intermediate System-to-Intermediate System (IS-IS) about a first source associated with the first source address;
 updating a second routing information database about the first source; and
 30 performing a lookup to obtain sites local interest for a first group associated with the first group address.
 4. The method of claim 3, further comprising:
 determining if any obtained sites have Internet Group Management Protocol (IGMP) oifs;
 sending a second multicast packet from a second source address to devices interested in the first group;
 discovering a router port at the first source; and
 snooping router port updates to discover a new oiflist.
 35
 5. The method of claim 4, further comprising:
 if the new oiflist is not null, notifying the first OTV site and an Intermediate System-to-Intermediate System (IS-IS) about the updated VLAN route entry; and
 sending an IGMP join message for sites with interest in the first group.
 40
 6. The method of claim 5, further comprising: storing the updated VLAN route entry to a multicast routing information database.
 7. The method of claim 1, comprising storing the updated VLAN route entry to a distribution memory.
 45
 8. A method comprising:
 installing a drop and punt route to entry cover a multicast range at a forwarding engine of a first local network, wherein installing the drop and punt route entry comprises installing the drop and punt route entry for the forwarding engine to:
 50

28

drop a first multicast data packet hitting the drop and punt route entry, and
 copy the first multicast data packet hitting the drop and punt route entry to a processor associated with the first local network;
 5 receiving, when a first multicast data packet hits the drop and punt route entry, the punted first multicast data packet at the processor;
 learning virtual local area network (VLAN) route information from the punted first multicast packet, wherein learning the VLAN route information comprises learning, from the punted first multicast data packet, a first VLAN information, a first source information, and a first group information;
 10 installing, based on the learned VLAN route information, a learned VLAN route on the forwarding engine located on the first local network through an overlay control plane of the first local network, wherein installing the learned VLAN route further comprises appending delivery information comprising a destination source information at the first local network and a destination group information at the local network to the learned VLAN route, wherein installing the learned VLAN route further comprises installing the learned VLAN route being configured to cause the forwarding engine to one of the following:
 15 drop, when no remote receivers exist, any multicast data packet hitting the learned VLAN route, and
 encapsulate, when one or more remote receivers exist, any multicast data packet hitting the learned VLAN route with a provider data delivery group information and send the encapsulated multicast data packet to the first local network; and
 20 distributing the learned VLAN route information to a plurality of remote layer-3 (L3) network on an independently operating client control plane.
 9. The method of claim 8, further comprising:
 receiving a second multicast data packet at an edge device, wherein the second multicast data packet hits the learned VLAN route; and
 copying the second multicast data packet to memory associate with the edge device.
 25
 10. The method of claim 9, further comprising: processing the copied second multicast data packet with a control plane component.
 11. The method of claim 10, further comprising:
 sending the VLAN route information from the copied second multicast data packet to the forwarding engine;
 dropping the copied second multicast data packet if no remote interested receivers exist; and
 sending the copied second multicast data packet if remote interested receivers exist.
 30
 12. The method of claim 11, further comprising:
 distributing the VLAN route information, using an overlay transport virtualization (OTV) to the remote interested receivers.
 13. The method of claim 12, further comprising: tracking the liveness of VLAN streams by periodically learning an incoming traffic rate.
 35
 14. A network device comprising:
 a memory storage;
 a processor coupled to the memory storage, wherein the processor is operative to:
 40 install a route entry associated with multicast traffic to into a first routing information database associated

29

with a forwarding engine of a service provider network, wherein the installed route entry causes the forwarding engine to:

drop a first multicast data packet hitting the route entry, and 5

punt the first multicast data packet hitting the route entry to a first processor associated with the service provider network;

notify client devices of the route entry for advertisement by an active source device over an overlay control plane; 10

wherein, when the multicast data packet hits the route entry, the first multicast data packet is punted at the first processor, wherein the first processor is configured to learn virtual local area network (VLAN) route information from the punted first multicast packet, wherein the VLAN route information comprises a punt flag that indicated to punt the punted multicast data packet to the first processor from a plurality of remote layer 3 (L3) network on an independently operating client control plane, and wherein the first processor is further configured to learn, from the punted first multicast data packet, a first VLAN address, a first source address, and a first group address; 15

install a learned VLAN route entry based on the learned VLAN route information, wherein the learned VLAN route entry is updated to append delivery information comprising a destination source address at a first remote L3 network and a destination group address at the first remote L3 network to the VLAN 20

30

route entry, wherein the learned VLAN route being configured to cause the forwarding engine to one of the following:

drop, when no remote receivers exist, any multicast data packet hitting a learned VLAN route, and encapsulate, when one or more remote receivers exist, any multicast data packet hitting the learned VLAN route with a provider data delivery group information and send the encapsulated multicast data packet to the plurality of remote L3 network; distribute the learned VLAN route entry; retrieve delivery group and delivery source information for the learned VLAN route entry; and maintain the delivery group and delivery source information in multicast data trees. 25

15. The network device of claim **14**, wherein the processor is further configured to:

determine interested local receivers by referring to a routing information database; and

handle join requests to the multicast data trees. 20

16. The network device of claim **15**, wherein the processor is further programmed to: advertise the learned VLAN route entry via an intermediate system-to-intermediate system (IS-IS).

17. The network device of claim **14**, wherein the network device is an overlay transport virtualization (OTV) network edge device. 25

18. The network device of claim **14**, wherein the processor is further configured to snoop via an Internet Group Management Protocol (IGMP). 30

* * * * *